NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

DYNAMIC EXPLORATION OF HELICOPTER RECONNAISSANCE THROUGH AGENT-BASED MODELING

by

Craig S. Unrath

September 2000

Thesis Advisor: Donald Gaver

Thesis Co-Advisors: John Hiles
Patricia A. Jacobs

Approved for public release; distribution is unlimited

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	September 2000	Master's Thesis

4. TITLE AND SUBTITLE
Dynamic Exploration of Helicopter Reconnaissance Through Agent-Based Modeling

5. FUNDING NUMBERS

6. AUTHOR(S)
Unrath, Craig S.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey, CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(maximum 200 words)*

This thesis uses Multi-Agent System modeling to develop a simulation of tactical helicopter performance while conducting armed reconnaissance. It focuses on creating a model to support planning for the Test and Evaluation phase of the Comanche helicopter acquisition cycle. The model serves as an initial simulation laboratory for scenario planning, requirements forecasting, and platform comparison analyses.

The model implements adaptive tactical movement with agent sensory and weaponry system characteristics. Agents are able to determine their movement direction and paths based on their perceived environment, attributes, and movement personalities. The model incorporates a three-dimensional aspect to properly simulate aerial reconnaissance. An integrated Graphical User Interface enables the user to create environments, instantiate agent propensities and attributes, set simulation parameters, and analyze statistical output.

The resulting model demonstrates the ability to represent helicopter reconnaissance behavior. It captures simulation summary statistics that illustrate enemy performance, helicopter performance, and logistical requirements. The model establishes an initial simulation tool to further explore Comanche operational requirements and planning for its Test and Evaluation phase.

14. SUBJECT TERMS
Multi-Agent System, Agent-Based Modeling, Helicopter Reconnaissance, Comanche, Adaptive Behavior, Modeling and Simulation

15. NUMBER OF PAGES 140

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

NSN 7540-01-280-5500 Standard Form 298 (Rev. 2-89)

Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

DYNAMIC EXPLORATION OF HELICOPTER RECONNAISSANCE THROUGH AGENT-BASED MODELING

Craig S. Unrath
Captain, United States Army
B. S., University of North Dakota, 1990

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN MODELING, VIRTUAL ENVIRONMENTS, AND SIMULATION

from the

NAVAL POSTGRADUATE SCHOOL
September 2000

Author: Craig S. Unrath

Approved by: Donald Gaver, Thesis Advisor

John Hiles, Thesis Co-Advisor

Patricia A. Jacobs, Thesis Co-Advisor

Rudy Darken, Academic Associate
Modeling, Virtual Environments, and Simulation Academic Group

Michael Zyda, Chair
Modeling, Virtual Environments, and Simulation Academic Group

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

This thesis uses Multi-Agent System modeling to develop a simulation of tactical helicopter performance while conducting armed reconnaissance. It focuses on creating a model to support planning for the Test and Evaluation phase of the Comanche helicopter acquisition cycle. The model serves as an initial simulation laboratory for scenario planning, requirements forecasting, and platform comparison analyses.

The model implements adaptive tactical movement with agent sensory and weaponry system characteristics. Agents are able to determine their movement direction and paths based on their perceived environment, attributes, and movement personalities. The model incorporates a three-dimensional aspect to properly simulate aerial reconnaissance. An integrated Graphical User Interface enables the user to create environments, instantiate agent propensities and attributes, set simulation parameters, and analyze statistical output.

The resulting model demonstrates the ability to represent helicopter reconnaissance behavior. It captures simulation summary statistics that illustrate enemy performance, helicopter performance, and logistical requirements. The model establishes an initial simulation tool to further explore Comanche operational requirements and planning for its Test and Evaluation phase.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

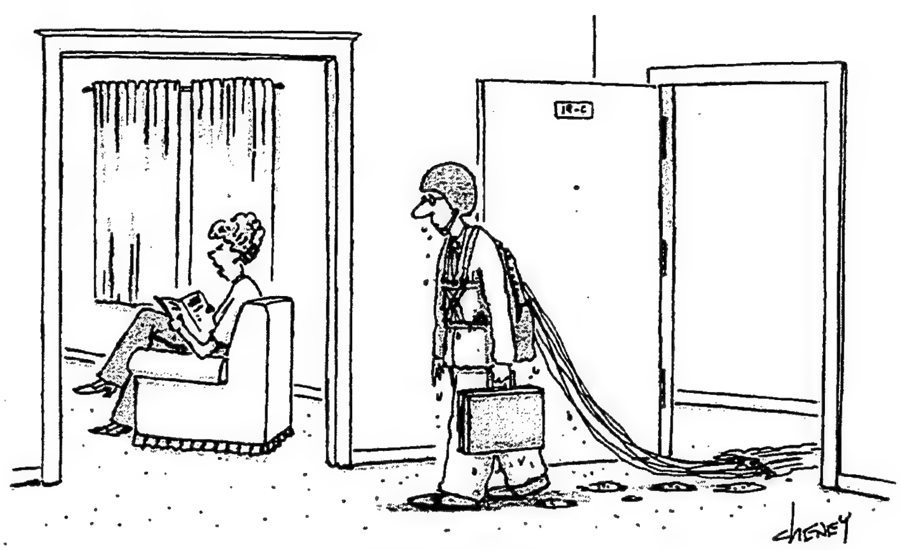
THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGEMENTS

I would like to express my sincere thanks to Dr. Donald Gaver, Dr. Patricia A. Jacobs, Mr. John Hiles, and MAJ Thom Crouch for their guidance, patience, and motivation. Without their broad knowledge in agent-based modeling, statistical analysis, and Comanche acquisition this research and model could not have been completed.

Additionally, I would like to thank my peers for their assistance and camaraderie during the challenging two years encompassing this degree and thesis research. Special thanks are extended to Tom Miller, Chris Buziak, and Mike Dickson. Your friendship and assistance in projects, homework, and computer coding made the achievement of this degree obtainable and enjoyable.

Finally, I would like to thank my loving wife Melissa and daughter Megan for their tremendous support during this challenging period in our lives. Melissa's devotion to our family and marriage made this degree and thesis a reality. Words alone can never express my overwhelming appreciation and gratitude. The following cartoon best sums up our past two years.



"How was your flight?"

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS

ACT	-	Air Cavalry Troop
ADA	-	Air Defense Artillery
AH	-	Attack Helicopter
COI	-	Critical Operational Issue
CRP	-	Combat Reconnaissance Patrol
DOD	-	Department of Defense
EMD	-	Engineering, Manufacturing, and Development
FAA	-	Forward Assembly Area
FARP	-	Forward Arming and Refueling Point
FDTE	-	Force Development Test and Experimentation
FLIR	-	Forward Looking Infra-Red
FM	-	Field Manual
FSE	-	Forward Support Element
GUI	-	Graphical User Interface
IOTE	-	Initial Operational T&E
ISAAC	-	Irreducible Semi-Autonomous Adaptive Combat
ITP	-	Integrated Training Program
KM	-	Kilometers
KPH	-	Kilometers Per Hour
KTPH	-	Knots Per Hour
LD	-	Line of Departure
LFTE	-	Live Fire T&E

LRIP	-	Low Rate Initial Production
LUTE	-	Limited User T&E
MAS	-	Multi-Agent System
MCCDC	-	Marine Corps Combat Development Command
METL	-	Mission Essential Task List
MRR	-	Motorized Rifle Regiment
MTOE	-	Mission Table of Organization and Equipment
NAI	-	Named Area of Interest
NOE	-	Nap Of the Earth
NPS	-	Naval Postgraduate School
OH	-	Observation Helicopter
ORD	-	Operational Requirements Document
PDRR	-	Program Definition Risk Reduction
RAH	-	Reconnaissance Attack Helicopter
ROS	-	Relief On Station
T&E	-	Test and Evaluation
TEMP	-	T&E Master Plan
TOC	-	Tactical Operations Center
TOS	-	Time On Station

I. INTRODUCTION

A. THESIS STATEMENT

A multi-agent system simulation in support of the Comanche acquisition cycle provides the combat developer and operational test community with an innovative modeling environment capable of demonstrating dynamic, tactical and system performance for scenario planning, requirements forecasting, and platform comparison analyses prior to the operational Test and Evaluation (T&E) phase.

B. MOTIVATION

The U.S. Army is developing the RAH-66 Comanche helicopter to replace an aging fleet of scout aircraft, and to fulfill cavalry mission requirements currently assigned to attack aircraft. Today's scout platforms represent 30-year-old technology that does not conform to current technological connectivity requirements, and is quickly becoming financially unsupportable due to budget constraints [TEMP, 1999]. The AH-64 Attack Helicopter is often used to fill corps and divisional reconnaissance mission voids. Although the newest Apache models are technologically advanced and lethal, the Apache is a large weapons platform that was primarily developed to fulfill the attack role by bringing an abundance of firepower to the battlefield. The Apache is not ideally suited to conduct the primary cavalry roles, such as reconnaissance. The Comanche will ultimately replace the OH-58D Kiowa Warrior as the primary aerial scout platform, and reassume cavalry missions currently covered by Apaches and Kiowa Warrior aircraft.

The Comanche is currently in the Engineering, Manufacturing, and Development (EMD) Phase of the Department of Defense (DOD) acquisition cycle. While in this

phase, material and combat developers, along with the contractor, will conduct tests and evaluations to assess the initially developed Comanche's operational performance status. Currently, operational test personnel do not possess any type of simulation tool to project the Comanche mission-specific performance or support requirements prior to test events. Often these projections are based on other historical test data or parametric analyses. Inaccurate estimates can add to an already financially burdened T&E process.

Reconnaissance is the primary cavalry mission identified in the Comanche Operational Requirements Document (ORD) [ORD, 1999]. Reconnaissance is an important method for acquiring early information about enemy forces and terrain conditions for a commander. The commander's ability to gain location information about the enemy, and so to direct early fires on that enemy is critical. The Comanche's advanced sensory capabilities and weaponry promise to be well suited to fulfill the general role of reconnaissance.

The tactical essence of aerial reconnaissance operations is difficult to capture through traditional discrete event and engineering simulation. Helicopters and enemy vehicles base movement on tactical desires and goals that are hard to replicate through steadfast rules. Additionally, helicopters react and move differently, based on how they perceive their enemy and the terrain.

The methodology behind multi-agent based modeling provides an innovative approach to model the tactical movement and interaction inherent to reconnaissance. This thesis is supported through the development of a model implementing Multi-Agent System (MAS) characteristics, the logic of the Irreducible Semi-Autonomous Adaptive Combat (ISAAC) model, and the tactical aspects of helicopter armed reconnaissance.

This thesis shows how this type of modeling can reasonably and suggestively represent tactical operations, such as helicopter reconnaissance. It additionally demonstrates the potential of this model to serve as an invaluable simulation tool for scenario planning, requirements forecasting, and platform comparison analyses prior to the Comanche's production and subsequent operational T&E phase, and field employment.

C. THESIS GOALS

The overall goals of this thesis are:

- Determine how to integrate the characteristics and adaptive behavior of MAS modeling with the tactical movement fundamentals of reconnaissance, specifically armed reconnaissance.

- Develop an initial modeling laboratory that demonstrates the successful implementation of reconnaissance through MAS simulation. Fully integrate the primary tactical aspects (movement, detection, and engagements), and vehicle attributes (station time, ammunition, sensors).

- Demonstrate model usefulness and potential through the output and analyses of summary statistics gathered from experimental simulation scenarios. Illustrate the model's ability to analyze tactical performance, assist in logistics forecasting, and potentially conduct helicopter platform comparisons in support of Comanche T&E.

- Provide future students with an initial model to build upon for the analysis of the Comanche's future T&E requirements, and to conduct more in-depth analyses.

- Provide students with an initial agent-based modeling environment in which to develop different types of MAS models in other areas of research interest.

D. THESIS ORGANIZATION

This thesis is organized into the following chapters:

- Chapter I: Introduction. Identifies the purpose and motivation for conducting this thesis research. Establishes the goals and objectives for this thesis.

- Chapter II: Background. Identifies how a simulation tool that models tactical operations could assist Comanche T&E; defines doctrine for aerial reconnaissance; and describes previous research in the field of adaptive MAS and agent-based modeling.

- Chapter III: Model Development. Describes the process, methodology, and major algorithms created during the development of the MAS implemented to model helicopter armed reconnaissance.

- Chapter IV: Model Analysis and Results. Shows and analyzes resulting summary statistics gathered from experimental model runs for various scenarios, agent propensities, and helicopter-platform attributes.

- Chapter V: Future Work and Conclusion. Discusses the model's potential for carrying out more advanced analyses, analyzing more advanced agent behavior, benefiting from the implementation of more enhanced operational integration, and better assessment of the Comanche's success at achieving its future requirements.

II. BACKGROUND

A. POTENTIAL SUPPORT OF THE COMANCHE HELICOPTER OPERATIONAL TEST & EVALUATION

The Comanche helicopter is currently being developed to replace the U.S. Army's aging fleet of light helicopters. The AH-1 Cobra and OH-58A/C helicopters have primarily transitioned to U.S. Army guard and reserve units, while the active duty OH-58D Kiowa Warrior technology is quickly becoming outdated. The U.S. Army is developing the Comanche to perform armed reconnaissance, security, and attack missions across the range of military distributed operations, while minimizing operational and support costs, conforming to digitization standardization, and extending the range of combined arms operations [TEMP, 1999].

On 4 April 2000, the Comanche successfully obtained milestone II approval and is presently in the EMD phase of the Department of Defense (DOD) acquisition cycle. One of the primary objectives of this phase is to demonstrate system capabilities through operational testing. During the Program Definition Risk Reduction (PDRR) phase of the acquisition cycle the T&E Master Plan (TEMP) was approved describing how operational T&E would be conducted during the EMD phase. Low Rate Initial Production (LRIP) is initiated during the EMD phase to support the proposed test plan. The objective of LRIP is to produce the minimum number of aircraft needed to sufficiently conduct T&E of the total system prior to full-rate production [5000.2-R, 1996].

The Comanche's initial LRIP quantity will be verified using eight production-representative RAH-66 Comanche aircraft. The primary purpose of operational T&E is

to determine whether systems are operationally effective and suitable for the intended use by representative users before production or deployment [5000.2-R, 1996]. The Operational T&E Plan of the Comanche program provides a means to assess the Comanche is operational effectiveness, and suitability, and survivability for use by operators, maintainers and support personnel [TEMP, 1999]. See Table 1 for depiction of proposed T&E dates for the Comanche helicopter during the EMD phase of the acquisition cycle. Ultimately the RAH-66 Comanche should extend the maneuver commander's battle space through reduced logistical requirements, extended imagery and weaponry, more efficient user operation, digitized connectivity, and unsurpassed mission versatility.

To validate these project capabilities, operational tests will place the Comanche in the hands of future users to evaluate the aircraft's ability to perform its required tasks in realistic situations. These users assess the Comanche's capability to perform the missions for which it was designed. The majority of these tests involve actual field-tests, with some factors supported through simulation. These test requirements create an enormous task for projecting logistic requirements and system performance.

Development of a simulation and modeling environment to explore and forecast requirements prior to the Force Development Test and Experimentation (FDTE) III, FDTE IV, Limited User T&E (LUTE), and Initial Operational T&E (IOTE) would serve as an invaluable tool during this critical phase of the Comanche development cycle.

UNCLASSIFIED				
Test Article	**Test Event**	**Quantity**	**Start Date**	**Source**
Prototype Aircraft	Prototype Testing	2	FY95	PDRR Contract
Engineering Design Simulator	CAP	1	FY01	EMD Contract
Comanche Portable Cockpit	FDTE I,II,III,&IV and LUTE	2	FY00-06	PDRR Contract
Simulated Crew Work Stations	FDTE II, III, IV, LUTE, IOTE	6	FY03-06	EMD Contract
Prototype Aircraft	Developmental testing EOSS User Survey	1	FY03	EMD Contract
Pre-Production Aircraft	LUTE , FDTE III	4	FY05	EMD Contract
Pre-Production Aircraft	FDTE IV, IOTE	8	FY-06	EMD Contract
Pre-Production Aircraft	LFTE	1	FY05-06	EMD Contract
ITP	ARTEP and IOTE	1 set	FY06	EMD Contract
Kiowa Warrior**	IOTE	8	FY06	FORSCOM
PSTB	PSTB Testing	1	FY95	PDRR & EMD Contract
Static Test Article	STA Testing	1	FY95	PDRR Contract
* LFTE Test article resource requirements for component tests are outlined in Part IV, Table 4-5; Full up LFTE will require one aircraft and repair parts which will be determined at a later date.** Kiowa Warrior requirements may be reduced if the approved Test and Evaluation Plan does not require a side-by-side force-on-force comparison IOTE				

Table 1. Comanche TEMP Test Article Matrix. From Ref. [TEMP, 1999].

Critical Operational Issues (COIs) are identified in the TEMP to set objective criteria for measuring Comanche LRIP performance during T&E. COIs are phrased as questions about the mission and operational effectiveness and suitability of the aircraft's associated systems and capabilities. The Comanche's primary COIs are [TEMP, 1999]:

- How well does the Comanche-equipped unit conduct operations compared to the baseline unit?
- How well does the Comanche-equipped unit achieve the Commander's sustained combat requirements?
- Does the Integrated Training Program (ITP) enable the acquisition of skills required to operate, maintain and support the Comanche?
- How survivable is the Comanche-equipped unit compared to the baseline unit while performing its assigned mission?

Within these COIs are specific criteria established to evaluate the Comanche's tactical capability, maintainability, logistical supportability, effect on personnel training, and survivability. The primary comparison platform (baseline unit) used for measuring performance improvement will be the OH-58D Kiowa Warrior. These COIs are used to obtain parameters for operational characteristics of the Comanche and evaluation scenarios. These parameters, contrasted with Kiowa Warrior performance parameters and evaluation scenarios, could serve as the data to extract and define marginal differences between the two systems.

This thesis proposes that modeling and simulation can play an integral role in assisting T&E personnel prepare and plan for future test scenarios. A fully developed modeling tool provides lead-time and insight for the FDTEs, LUTE, and IOTE. A model

could facilitate operations plan staffing by T&E personnel, and help establish common goals toward the overall T&E plan. This thesis is supported through the development of an initial agent-based modeling tool that captures some of the previously discussed Comanche T&E requirements related to the primary tactical mission role of reconnaissance.

B. RECONNAISSANCE

The U.S. Army officially defines reconnaissance as a mission undertaken to obtain information by visual observation, or other detection methods, about the activities and resources of an enemy, or about the meteorological, hydrographic, or geographic characteristics of a particular area [FM 17-95, 1996]. Reconnaissance is primarily a mission used by a commander to gain information about an enemy force and the terrain that will be encountered in future operations.

There are four general methods for performing reconnaissance: aerial, mounted, dismounted, and armed reconnaissance [FM 17-97, 1995]. These four methods relate to the many different types of units and equipment tasked to perform this mission. Reconnaissance assets range from infantrymen, to tanks, to unmanned aerial vehicles. This thesis focuses on aerial reconnaissance and armed reconnaissance techniques; more specifically helicopter armed reconnaissance.

Aerial reconnaissance via helicopters provides numerous advantages and capabilities to a ground maneuver force. Helicopters are able to provide earlier warning of enemy activity, secure flanks during movement, speed the rate of movement, and inflict early damage and cause chaos amongst enemy forces. For these reasons, the use

of helicopters to conduct reconnaissance is an integral part of the U.S. Army's combined arms team.

Reconnaissance operations consist of four types: route, area, zone, and reconnaissance in force. Although helicopters are capable of accomplishing all four of these missions, this thesis will explore and model only zone reconnaissance. Route and area reconnaissance deal primarily with gathering information specific to a location or feature. Information gathering during zone reconnaissance is directed on the features and enemy within a bounded sector. Zone reconnaissance missions are assigned when enemy situations are vague or terrain information is unavailable [FM 17-95, 1996].

The weaponry and sensory equipment associated with today's helicopters has changed the capabilities of U.S. Army reconnaissance operations. Definitions for the traditional types of reconnaissance no longer fully capture the total helicopter-mission spectrum. No longer are reconnaissance teams only tasked to gather information and proceed with stealth. Today's reconnaissance helicopters can detect enemy activity at greater distances and then destroy those forces without becoming decisively engaged.

These advanced capabilities create some doctrinal inconsistency when defining this aggressive method of reconnaissance using force. Zone reconnaissance operations with mission tasks to find and destroy are often called armed reconnaissance, reconnaissance in force, reconnaissance by fire, or movement to contact. None of these mission names doctrinally define this technique correctly. Nonetheless, this new role has become a primary mission for cavalry and attack aviation units when setting the conditions for future operations. In accordance with FM 1-114, reconnaissance by fire is the doctrinally correct name of the zone reconnaissance mission incorporating detection

and destruction of the enemy. But, to maintain consistency throughout this thesis, better conform with Comanche T&E documentation, and assist the reader's conceptual understanding, it will be referred to as armed reconnaissance.

Armed reconnaissance missions are common missions assigned to attack and cavalry aviation units within light and heavy divisions. Armed reconnaissance is often an integral task in these types of unit Mission Essential Task Lists (METL). The success of these missions is critical to the future maneuver operation for the division or brigade/task force. The ability of Aviation units to surprise, find, report, and destroy the enemy early often shapes the outcome and maneuver direction of follow-on ground forces.

Armed reconnaissance missions attempt to place fires on positions the enemy is suspected of occupying in order to disclose enemy positions, intent, and capabilities. A commander uses this type of mission given any variation of the following conditions [FM 1-114, 2000]:

- Situation meets strict engagement criteria
- Time is critical
- Encountering obstacles that could be over-watched by an enemy
- An enemy position is suspected
- Enemy locations are known

The armed reconnaissance mission is usually assigned to aviation cavalry squadrons or attack battalions. These units often further delegate the mission to their internal Air Cavalry Troops (ACT) or attack companies, dependent on the size of the terrain and required amount of coverage time. Ultimately, ACTs and companies assign

these missions to teams of two aircraft to cover the defined sector for specified periods of time until relieved by follow on teams (called Relief On Station, ROS), sustained damage, or require immediate critical maintenance.

Mission orders consist of very detailed information, but primary mission information consists of a mission type (like armed reconnaissance), suspected enemy positions or Named Areas of Interest (NAI), and responsible time of coverage. Teams generally determine their own methods of terrain coverage, routes, and actions on contact. The possible situations during these types of missions are numerous; therefore Commanders entrust teams to employ previously trained team tactics and procedures. Teams generally base their movement through a sector on the terrain, cover, and enemy within the area, while ensuring total coverage of an area with respect to the Commander's thoroughness criteria. Orders also contain rules of engagement that direct how teams will react to enemy contact. During a mission, situation reports are continuously provided to Commanders to inform them of current status and developing situations.

Today these units and teams are equipped with OH-58D Kiowa Warrior or AH-64A Apache helicopters. Until the middle 1990's, traditional cavalry operations, like reconnaissance, were conducted with combinations of AH-1 Cobras (attack) and OH-58 (scout) helicopters. Aviation restructuring initiatives throughout the active duty divisions have since replaced these aging platforms with modern aircraft like the Kiowa Warriors and Apaches.

By the Mission Table of Organization and Equipment (MTOE), "true" cavalry units (units whose METLs consist primarily of cavalry operations), are equipped with the Kiowa Warriors. The Kiowa Warrior brought a tremendous improvement to

reconnaissance capabilities versus the AH-1s and OH-58s. Its target acquisition power, weaponry, and small profile make it a very good platform for the information gathering aspects of reconnaissance operations. But, the Kiowa Warrior's endurance and power limitations and aging technology hinder its ability to effectively perform the very critical reconnaissance mission, with or without armament.

Limitations concerning endurance and weapons basic loads prevent the Kiowa Warrior from effectively bringing firepower to the battlefield under certain conditions. These limitations often require increased Forward Arming and Refueling Point (FARP) rotations when using Kiowa Warriors for direct fire missions like armed reconnaissance [FM 1-114, 2000].

Given these limitations, modifications must be made to the Kiowa Warrior profile, or Apache units must be used to conduct the armed reconnaissance mission. Although the Apache brings an abundance of firepower to the battlefield, it is primarily an attack aircraft, not a low-profile reconnaissance platform. Therefore, a gap in mission capability exists because of inability to match the best aerial platform to the armed reconnaissance type of mission. The U.S. Army is developing the Comanche to fill these types of mission inefficiencies. The Comanche's ultimate objective is to possess the necessary sensory and weaponry capabilities, along with stealth and low profile, to better fulfill the requirements of cavalry operations such as armed reconnaissance.

C. ADAPTIVE, AUTONOMOUS AGENTS & MULTI-AGENT SYSTEMS

The previous description of armed helicopter reconnaissance lends itself well to the concepts of agent-based modeling and adaptive, autonomous agent behavior. Many of the operations, interactions, and tactics concerning the armed reconnaissance mission

involve difficult cognitive and reactive responses not easily captured through traditional modeling techniques. Helicopters and enemy vehicles rarely follow exact routes and directions when driving toward a goal or navigation objective. MAS modeling provides a more realistic perspective to study reconnaissance performance in a decentralized setting and overcome its poor relationship with centralized and discrete maneuvering.

A decentralized, adaptive, MAS better captures the actual movement of reconnoitering helicopters and maneuvering enemy vehicles. Mitchel Resnick cites numerous examples where society accepted the centralized solution to a problem or phenomena, only to find that the underlying outcome was the result of decentralization, not centralization [Resnick, 1994]. Resnick's real life examples include long-standing governments, nature, and even industry that show how the centralized theory is often perceived and poorly applied. Although reconnaissance is not a centralized system, many previous military models use centralized methodology to simulate reconnaissance movement behavior. MASs distinguish themselves from traditional modeling techniques by emphasizing the interactions and adaptability of the elements being studied [Ferber, 1999].

The primary MAS elements used to represent real-world physical entities are called *agents*. Ferber provides descriptive characteristics that make up an interactive agent. These are the agent characteristics adopted for implementation into the model of this thesis. Some of the attributes Ferber lists as characteristics needed to qualify an entity as an agent include the following [Ferber, 1999]: the agent must be able to act within an environment given a set of resources; agents are driven toward their goals by a function of their propensities; agents can sense their environment within prescribed

limits; and agents behave in a manner that best suits their objectives while monitoring resource levels and adjusting their intentions with respect to how they perceive their environment. This last characteristic is often called autonomous behavior. No longer does a simulation require user input to direct and govern agent movement and decisions. Rather, the agent parameters and propensities are set, allowing the agent to conduct movement independent of user intervention.

Given these agent characteristics, Ferber presents two methodologies for assigning intelligence: cognitive and reactive [Ferber, 1999]. Cognitive agents have preset intentions to drive their actions toward their objectives. Under this definition the agents are already intelligent. They possess the rules necessary to deal with any situation confronted. Reactive agents respond according to the information sensed from the environment. They do not possess other perceived information on which to base their future decisions. Helicopter reconnaissance applies to both of these types of agents. Therefore, limiting agent intelligence to merely one agent type may hinder the model's ability to capture aerial reconnaissance. The MAS developed in support of this thesis implements both types of agent characteristics.

Agents are one type of the primary MAS elements. The remaining primary MAS elements, in accordance with Ferber's definition, include: the environment, objects, relations, operations, and laws [Ferber, 1999]. The environment means the physical space that contains the entire system. The objects, such as terrain features, are situated, passive elements within the system. They differ from agents due to their inability to interact or adapt. Agents are able to manipulate objects, and objects can impact agent performance. Agents are objects, but objects are not necessarily agents. Relations serve

as commonalties to group agents. Operations are what give agents the capability to manipulate objects and other agents. Finally, laws are what Ferber uses to portray how the world reacts to the attempted modifications of the system. Given Ferber's explicit and concise definitions of these elements, it becomes much clearer how a MAS and adaptive, agent-based simulation could be tailored to model helicopter reconnaissance.

According to Ferber, a system like helicopter reconnaissance via two-ship teams relates best to a MAS-level called the micro-social level. In this (most commonly researched) level of organization, researchers emphasize individual agent interaction with respect to specific relations of a small number of agents. Although the study of reconnaissance at higher levels of organization is possible, the goal of this thesis it to build an initial MAS that revolves around the basic reconnaissance elements of helicopter teams and enemy combat vehicles.

D. IRREDUCIBLE SEMI-AUTONOMOUS ADAPTIVE COMBAT (ISAAC) MULTI-AGENT BASED MODEL OF LAND COMBAT

ISAAC is a skeletal agent-based model developed to explore individual ground combat as a Complex Adaptive System (CAS) that possesses many of the elements and characteristics of a MAS discussed above. Dr. Andrew Ilachinski developed ISAAC for the Commanding General, Marine Corps Combat Development Command (MCCDC) in 1997. This research was sponsored by the U.S. Marine Corps to study the applicability of this new concept to land warfare [Ilachinski, 1997]. Many of the agent-based, modeling aspects explored and developed within the ISAAC framework relate to the general characteristics of helicopter reconnaissance. Adaptive movement behavior incorporated within ISAAC could be applied to a helicopter reconnaissance MAS.

Conventional combat models have primarily been based upon the Lanchester Equations created in 1914 [Lanchester, 1914]. Ilachinski modeled land combat as a CAS. He viewed combat as a nonlinear, dynamic system, where many semi-autonomous agents interact and adapt to a continuously changing situated environment. To this day, the simplistic Lanchester attrition equations still continue to be applied to models that simulate modern warfare, with little or no regard to cognitive and adaptive aspects. Ilachinski, amongst others, felt that Lanchester Equations unsuccessfully represented the autonomous and adaptive tactical operations of today's small unit Marine Corps tactics. Ilachinski developed ISAAC to provide the Marine Corps with a different perspective when studying the interaction and adaptability of their modernized small-unit forces [Ilachinski, 1997].

ISAAC implements ISAACAs (ISAAC Agent) as agents to represent low-level combatants, such as individual infantrymen. These agents then adapt to the situated environment by responding to local information. Agent decisions are decentralized and driven solely by the personality propensities for each individual ISAACA. Their movement is adaptive and their decision-making methodology is consistent with command levels of decision-making [Ilachinski, 1997]. See Figure 1 for a screen-shot of the entire ISAAC simulation display.

ISAAC's situated environment consists of a two-dimensional grid system. The two types of agents (red and blue ISAACAs) are able to occupy any one of these grid positions. See Figure 2 for a detailed depiction of the ISAAC battlefield. The overall driving goal of these agents is to capture the opponent agent's flag in the opposite corner

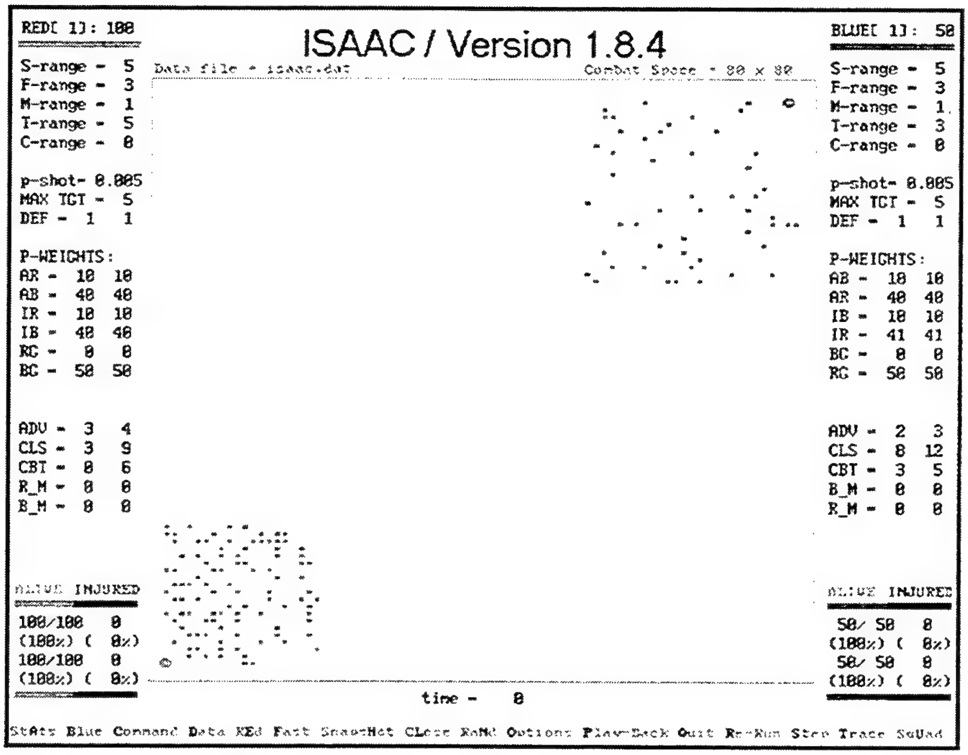


Figure 1. Main ISAAC screen. From Ref. [http://www.cna.org/isaac/sampscrn.htm].

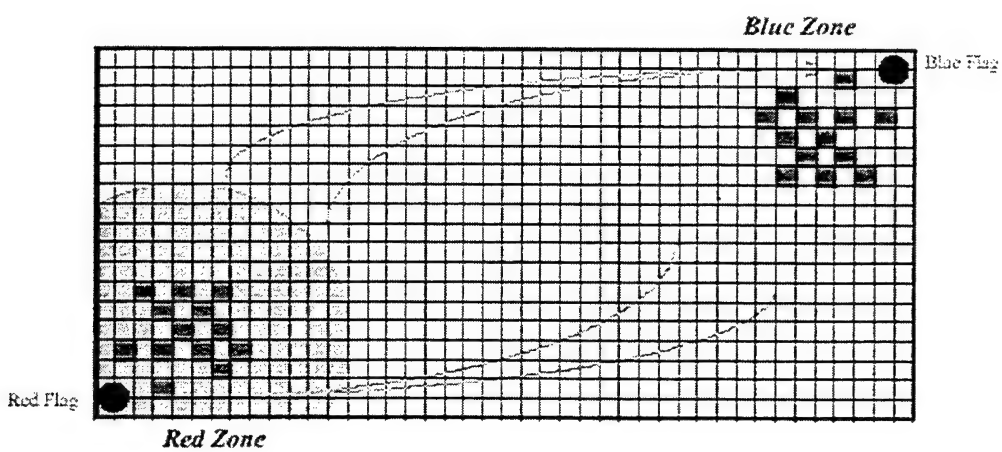


Figure 2. Putative two-dimensional "Combat Battlefield" in ISAAC. From Ref. [Ilachinski, 1997].

of the screen. During a simulation run the ISAACAs maintain one of three health statuses: alive, injured, or dead. Additionally, ISAACA's are equipped with various range characteristics for their sensor, shooting, threshold, movement, and communications. These ranges have various effects on what information is sensed by the ISAACAs and how far they can move. The injured status reduces an agent's shooting and movement range [Ilachinski, 1997].

One of the most interesting aspects of ISAAC is the implementation of adjustable personality vectors for the ISAACAs. These vectors consist of six separate propensities for each ISAACA. These personal propensities solely characterize the desired movement direction of the agents. The six elements that make up an ISAACA's personality or, intention to move towards, include: alive friendly, alive enemy, injured friendly, injured enemy, red flag, or blue flag. The user is able to adjust these propensities thus enabling the ability to create a myriad of personalities and adaptation patterns during simulation [Ilachinski, 1997]. This concept of defining a personality vector to drive adaptive agent movement became one of the primary innovations adopted into the helicopter reconnaissance simulation in support of this thesis.

The ISAACA personality vector is then integrated into what is called a penalty function. This penalty function serves as a mathematical calculation to determine the best future movement location for the ISAACA given its previously set personality. Each possible movement location is input into the penalty function to assign it a numerical value. The grid location resulting in the smallest numerical value (penalty) is chosen as the best move for that ISAACA. This location best satisfies the ISAACA's movement desires given all possible movement locations with respect to his sensor and movement

range. A similar form of this penalty function calculation was adapted to the helicopter reconnaissance simulation described in this thesis.

Figure 3 depicts an example move for a red ISAACA located in the center. The area of grid squares defines the ISAACA's sensible area. The squares immediately surrounding the ISAACA and its current location define the viable movement locations for the ISAACA. Once the sensed data are collected concerning nearby agents and distances to both flags, the penalty function is calculated for each of the possible nine moves (depicted in gray). The square giving the lowest value is chosen as the next move.

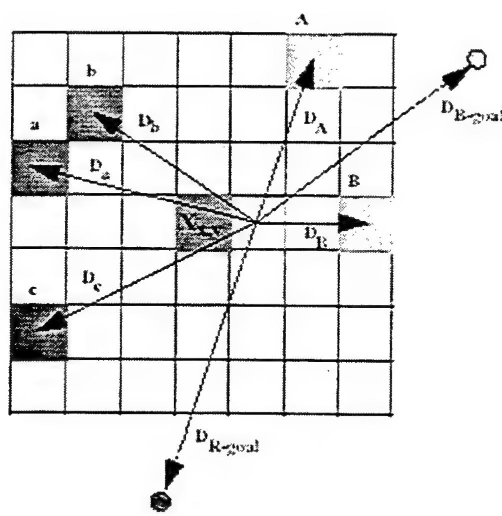


Figure 3. Sample penalty calculation.
From Ref. [Ilachinski, 1997].

Dr. Ilachinski's modeling methods create intelligent agents that adapt to their environment, rather than only act on discrete events. This thesis proposes that the ISAAC concept of adaptive, agent-based movement be applied to the study of helicopter reconnaissance. With a model that implements advanced target acquisition, advanced weaponry, and a third dimension to represent helicopter flight altitudes, it is possible to

capture the essence of an armed aerial reconnaissance mission. With the addition of specific application parameters, like those related specifically to helicopter reconnaissance, the ISAAC methodology of modeling can potentially produce future insights and new perspectives for projecting T&E requirements and performance during the Comanche helicopter acquisition life cycle.

THIS PAGE INTENTIONALLY LEFT BLANK

III. MODEL DEVELOPMENT

A. GENERAL

The background material presented in the previous chapter serves as the basis for developing a MAS model to explore helicopter armed reconnaissance. Doctrinal helicopter reconnaissance and MAS methodology are integrated to explore helicopter reconnaissance performance during T&E scenarios that are part of the acquisition cycle process. The following sections provide a broad explanation of the algorithms, methods, and data used to build the model framework. For more in-depth insight into the model, the reader is encouraged to further analyze the model's computer code. This model does not completely encompass helicopter reconnaissance. It was developed as a prototype proof of principle to initiate work in this area, and to establish a "virtual, agent-based, simulation workspace" for future development.

B. ENVIRONMENT

A user accesses the model program by way of a Graphical User Interface (GUI). The GUI allows the user to create a new environment or load a previously developed and saved environment. The environment is instantiated from within the main *ReconSim* Java class. Specifically, the environment establishes a new piece of terrain in which a simulation can be created for model execution. The environment creates a common area and visual display in which both types of agents will interact.

First created from the environment class is a *Map* object. Although the map manipulation methods are not located within the *Environment* class, access to the map object can be obtained through getter and setter methods. When a *new* map is created, a

light brown, desert-like background is displayed that is capable of sensing user mouse clicks. This initial display allows the user to click anywhere within the brown area to identify the location at which to place terrain and agent objects.

All vectors possessing the different types and status conditions of agents are also maintained within the *Environment* class. The vectors are data containers holding common agents such as the red (enemy), blue (helicopter reconnaissance teams), dead, and re-supplying agents. There are also vectors for holding the red agent re-supply caches and remaining checkpoints of the reconnaissance route for blue teams conducting Relief On Station (ROS). Additionally, once a blue team determines the need for relief, the common rally point for taking over the reconnaissance is set and retrieved from the environment class. These concepts are discussed in greater detail throughout this Chapter.

The only *paint* method used throughout the program is also located within the *Environment* class. The *paint* method is the common Java method used to draw the visual graphics display. The *paint* method first calls the *draw* method within the *Map* class to display the brown map terrain. If the user is opening (*load*) a previously saved environment, the objects and agents from that environment will also be displayed. Following the display of the map, *iterators* are used to retrieve each element from the vectors described above and position them according to their appropriate grid position with respect to the map.

It should be noted that computer screen graphics do not utilize the same (0,0) starting intersection as the standard geometric (x,y) layout. On a computer screen, (0,0) is the upper left hand corner. The positive y-axis runs from top to bottom along

the left side, and the positive x-axis runs from left to right along the top. This is important when interpreting (x,y) locations displayed in the *agent dialog box* (this dialog box is explained later). Figure 4 shows an example of two agent locations, the extreme corners, and the screen directions for the positive x and y-axes.

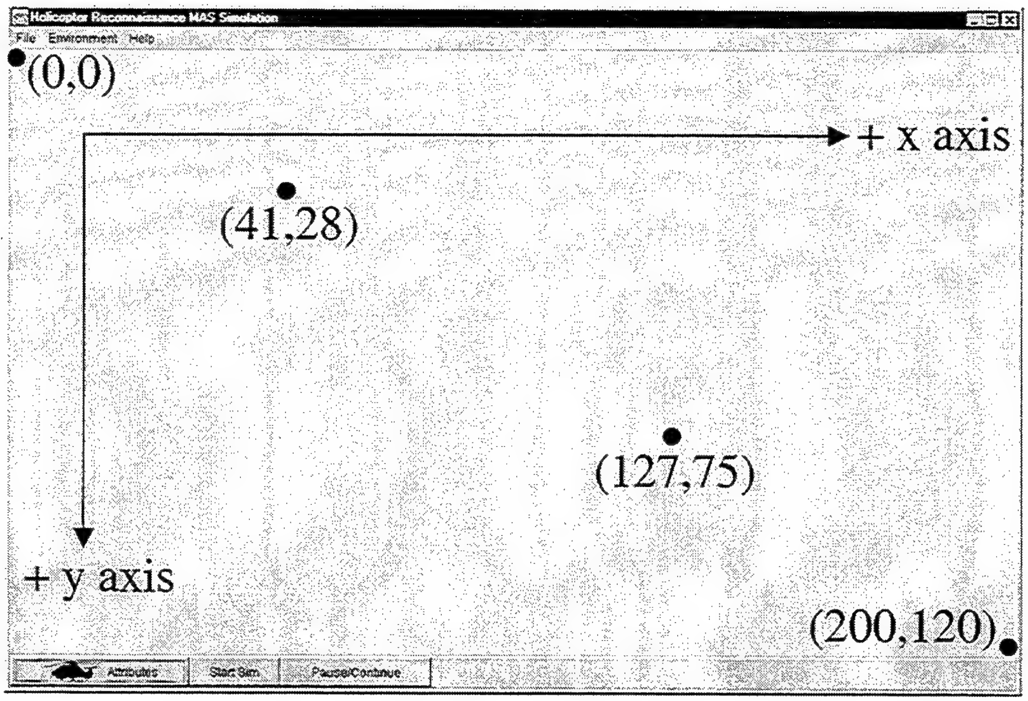


Figure 4. Environment orientation.

A *clear* method is also located in the *Environment* class. When called, the *clear* method completely erases the visual display leaving a white background, indicating to the user that there is currently no terrain map instantiated for developing a scenario.

C. MAP AND TERRAIN FEATURE OBJECTS

The *Map* class of the model serves as the graphical link between objects and the environment. A map object gives meaning to the pixels, objects, and colors positioned

inside the environment GUI window. It provides a proportional landscape to build and display the simulation environment.

The creation of the map starts by taking the pixel dimensions input from the creation of the environment and GUI panels. See Figure 6 for a visual display of the following dimensions. The pixels input for this model were hard-coded with the intent of portraying a map size that is doctrinally realistic with respect to a typical battalion area of operations. Pixel division by five was used to create a map consisting of 200 by 120 units (1000 by 600 pixels). With pixels representing 50 by 50 meters, the modified map resulted in map squares of 250 by 250 meters. Therefore, the resulting graphical dimensions of the map are 50 KM left to right (west to east), and 30 KM top to bottom (north to south).

The five by five pixel squares in this program represent what are commonly referred to as grid squares on military maps. These grid squares each possess a five-element array data structure. The five elements within this array consist of the applicable elevation, cover/concealment, agent occupation, maneuverability, and color information for each of the 12,000 grid squares of the map. The array information for a map square is constant throughout the entire area of a map square.

Elevation is used to represent the actual ground elevation of terrain, ground vehicles (red agents), and flight level of helicopter teams (blue agents). Elevation is not implemented in a true three-dimensional, graphical sense; instead it is represented by an integer value from one to six. The initial brown background displayed upon instantiation of a *new* environment represents the base elevation of one. These integers have no fixed relation to any particular height metric, but can be 100-meter increments, for example.

The proper use and implementation of elevation are the critical factors in this type of model, not the relation to real terrain relief. Eventually this model could be integrated with real three-dimensional terrain databases (discussed in Chapter 5).

There are various types of terrain features that can be displayed once a new environment is created. By clicking anywhere on the brown background, a pop-up window (see Figure 5) will be displayed where the user can select a terrain feature to be placed at the location just clicked with the mouse. The terrain features include: a mountain, hill, east-to-west ridge, north-to-south ridge, large covered area, and a small covered area.

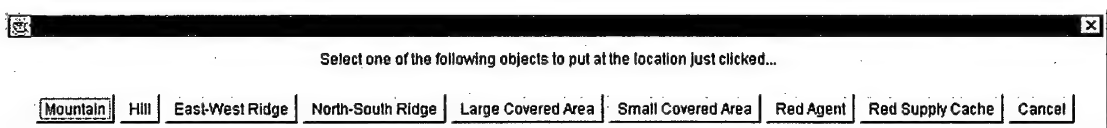


Figure 5. Object selection panel.

Placing a terrain feature at a location changes various values of the array elements associated with the surrounding map squares of the location just clicked. All terrain features are associated with different color representations. For features that represent changes in elevation such as mountains, hills, and ridges, six different shades of brown are used to depict the changes in elevation (see Figure 6).

For example, a mountain's elevation span includes all six elevation levels (1 - 6). Therefore, a mountain is positioned and displayed with all six colors. Whereas hills and ridges only range from elevation levels one to four. Selecting a large or small covered

area will not change the elevation level, but does change the array element for cover from zero to one, indicating that the area possess cover and concealment.

The green color associated with cover is used to represent vegetated areas as depicted by actual maps. Additionally, vegetation colors have a transparent effect and terrain features interlay or meld their elevations when placed on top of each other. For example, if a covered area is positioned over a terrain feature, the terrain contours underneath the vegetation will still be visible due to the transparency effect. If a hill is placed on top of a mountain, the mountain is not replaced; rather the higher of the two elevations has precedence and remains visible. See Figure 6 for a visual display of these explanations.

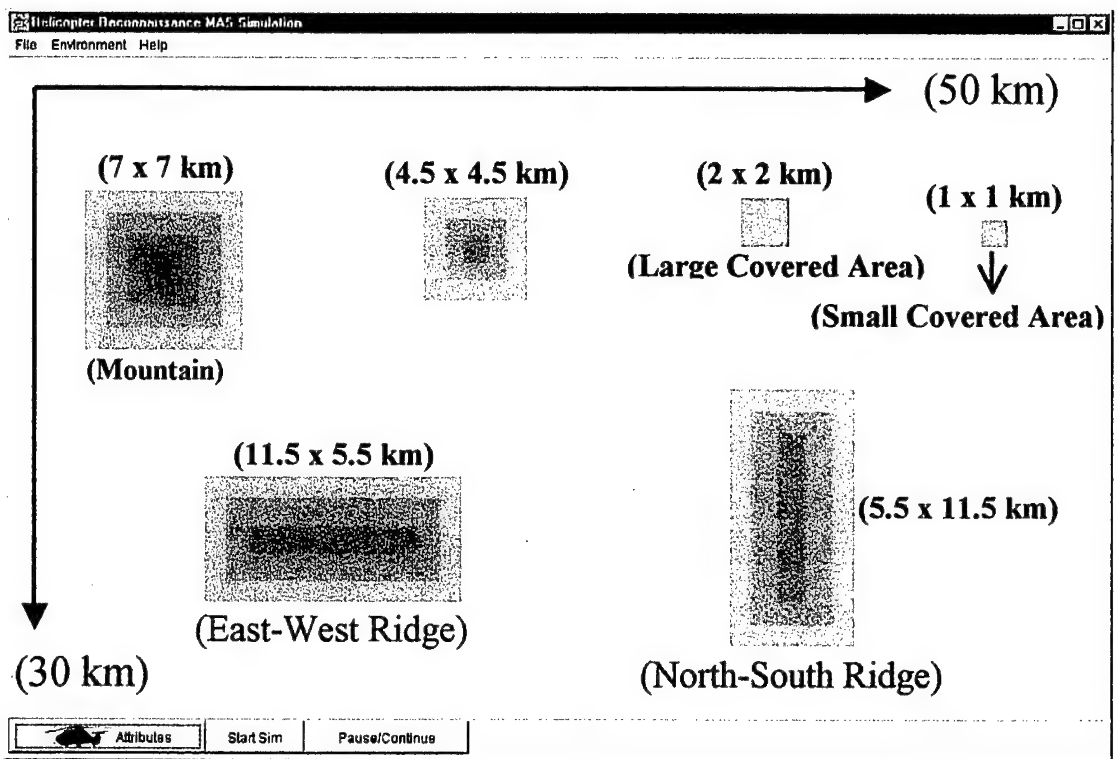


Figure 6. Terrain feature objects.

Each of the terrain features explained above possesses their own classes within the computer code for this program. The band of elevation and area coverage for each colored area within a feature varies amongst the features. For more explicit details of terrain feature creation and coverage, the reader is encouraged to see the computer code for each applicable object. These terrain features should not be thought of as obstacles or barriers within the environment. They depict and contain changes in elevation and cover that provide information to agents. The agents intelligently use this map information to accommodate their map reading and develop their movement strategies that will be discussed later.

D. AGENTS

Agents are the objects, which represent the interactive entities that operate within the artificial environment explained above. Agents actually sense their environment and intelligently adapt their actions according to their characteristics, sensed enemy, system attributes, and movement propensities. The agents chosen to represent the primary players of a reconnaissance mission scenario are tandem helicopter reconnaissance teams (friendly), and common Soviet military vehicles found in various doctrinal Soviet organizations (enemy). Blue circles with black perimeters are used to represent the helicopter reconnaissance teams, and red circles with black perimeters are used to represent the enemy vehicles.

Separate Java classes are used to create the two types of agents. A parent agent class is used to contain most variables common to both types of agents, and the separate child classes are used for specific characteristics not common to both types of agents. Some of the more common agent characteristics shared by both red and blue agents

include: (x, y) grid location, agent color, endurance or available Time On Station (TOS), movement speed, and statuses. Variables such as elevation, reconnaissance-specific methods, objectives, and routes are not common between agent types. Therefore, many of these data are maintained within the specific class of that type of agent when it is instantiated.

Many of the attributes pointed out above are input by the user of the model. The combination of these different individual characteristics leads to one of the most insightful capabilities of this simulation. Different combinations can often result in many different outcomes and agent performances during a simulation run. The characteristics for each agent are broken down into two different types, attributes and propensities. Attributes apply more to the discrete aircraft, team, and vehicle performance capabilities, whereas propensities apply to the tactical movement tendencies of the agents. These agent characteristics are input by the user through Java panels that contain sliders for each attribute and movement propensity (see Figures 7, 9, and 10).

1. Red Agent Attributes

Red agents are first created by mouse-clicking on the desired location within the map for placement, followed by selecting select *Red Agent* in the object selection window (see Figure 5). A panel is then displayed for setting the characteristics of a red agent (see Figure 7). Red agent characteristics consist of five attributes and four movement propensities. Figure 7 shows how the slider bars are arrayed on the Java panel.

The first red agent attribute enables the user to set the type of vehicle that the red agent will represent. There are five available choices, which represent three different categories of military equipment: armor, Air Defense Artillery (ADA), and lightly

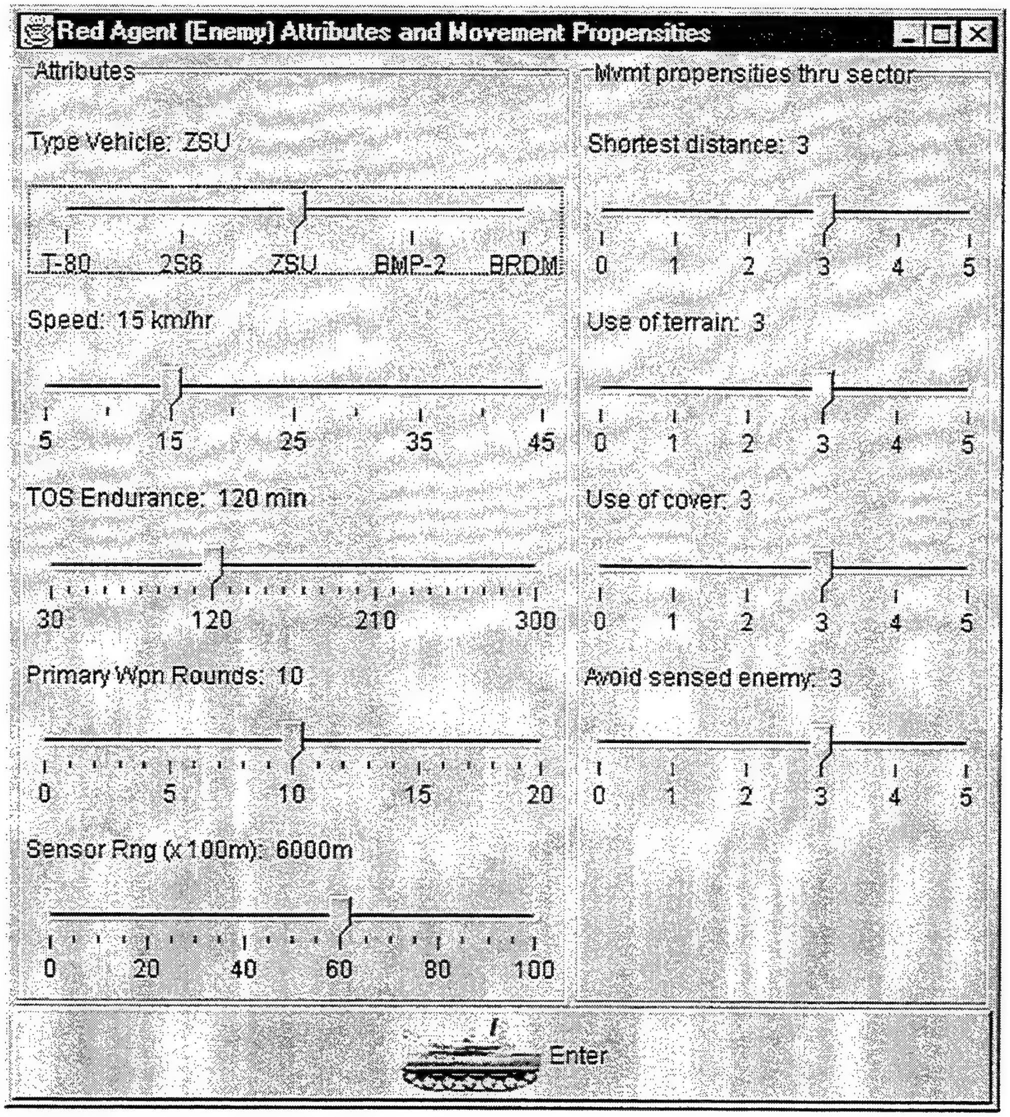


Figure 7. Red agent instantiation slider panel.

armored infantry fighting vehicles. The various types of vehicles available for selection within these categories include: the T-80 main battle tank (armor), 2S6 and ZSU-23-4 (ADA), and BMP-2 and BRDM-2 (infantry fighting vehicles). By choosing one of these vehicle types, the primary weapon system for engaging aircraft is set. This vehicle type establishes the appropriate percentages for the vehicle's associated primary weapon and

own survivability probability of hit and conditional probability of kill given hit. These percentages were obtained from the U.S. Army's combined arms simulation trainer, JANUS, version 7.06dc [JANUS, 1999].

The second red agent attribute is the agent's speed. The available choices for this attribute range from five to 45 kilometers per hour (KPH) with five KPH increments for each type of vehicle. The user should have some sense of tactical vehicle movement speeds when assigning speeds to the various vehicle types. The KPH metric is used because agent movement is proportionally related to the map dimensions previously discussed. The speed set for an agent will establish the distance covered by an agent for each simulation event-step.

The third red agent attribute is the agent's endurance time or TOS within the sector before needing refueling. This attribute ranges from 30 to 300 minutes (one-half to five hours) in ten-minute increments. Longer TOSs enable agents to spend more time moving toward their objective before having to move toward and park at re-supply cache sites.

The primary weapon rounds attribute allows the user to specify how many rounds are available to the agent for engaging opposing force agents (blue agents) before the agent must travel to a re-supply cache to rearm. This attribute ranges from zero to 20 for all five primary weapons systems associated with the five types of red agent vehicles in this model. Ammunition values for red agents should be understood as engagement credits rather than "rounds". For example, a red agent with ten available engagement credits can shoot at enemy agents ten discrete times, no matter if the weapon is a missile

or gun system. As with TOS, red agents possessing fewer engagement credits may result in them traveling toward and rearming in re-supply cache sites more often.

The sensor range is an attribute that allows the user to set the maximum detection range of the red agent's target acquisition system. The red agent's ability to successfully detect opposing blue agents depends on this attribute, along with other agent characteristics. This value ranges from zero to 10,000 meters in 500-meter increments. The other uses of this attribute are covered in more specific detail in the target detection section.

2. Blue Agent Attributes

Blue agents possess many of the same attributes as discussed above for red agents, with some minor differences. Blue agent attributes are broken into two categories versus one for red agents. These categories include: team attributes and helicopter attributes. These agent characteristics are also input by the user through a Java panel that contains sliders for each characteristic (see Figures 9 and 10). The user brings up this window by pressing *Attributes* (black helicopter silhouette) button at the bottom of the environment window, followed by selecting the helicopter platform type (see Figure 8).

The team attributes are characteristics that are common to both agents within a two-ship team. Helicopter attributes are values specific only to one of the elements within the two-ship helicopter team. Therefore, when a team is instantiated, it is credited with double the amounts of ammunition to represent both aircraft that make up a blue team. A blue team agent is instantiated once the user is satisfied with the characteristic settings and presses the *Enter* button at the bottom of the panel. Note that both helicopters in a team are of the same type.

Since the vehicle system used by the blue agents is already known (helicopters), system selection is not an option. But, given that this simulation was created to potentially support comparison capabilities between the U.S. Army's future helicopter reconnaissance platform and its predecessor, the user is able to select the Comanche or Kiowa Warrior helicopter type. Figure 8 shows the helicopter type selection box presented to the user when the helicopter *Attributes* button is pressed. Figures 9 and 10 show the attribute and propensity panels for the Comanche and Kiowa Warrior respectfully. The system differences represented by these panels are the endurance, missile loads, and gun type/rounds.

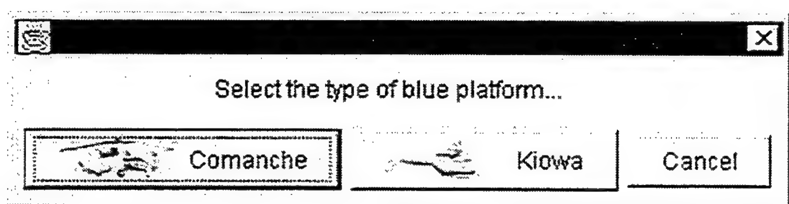


Figure 8. Helicopter type selection box.

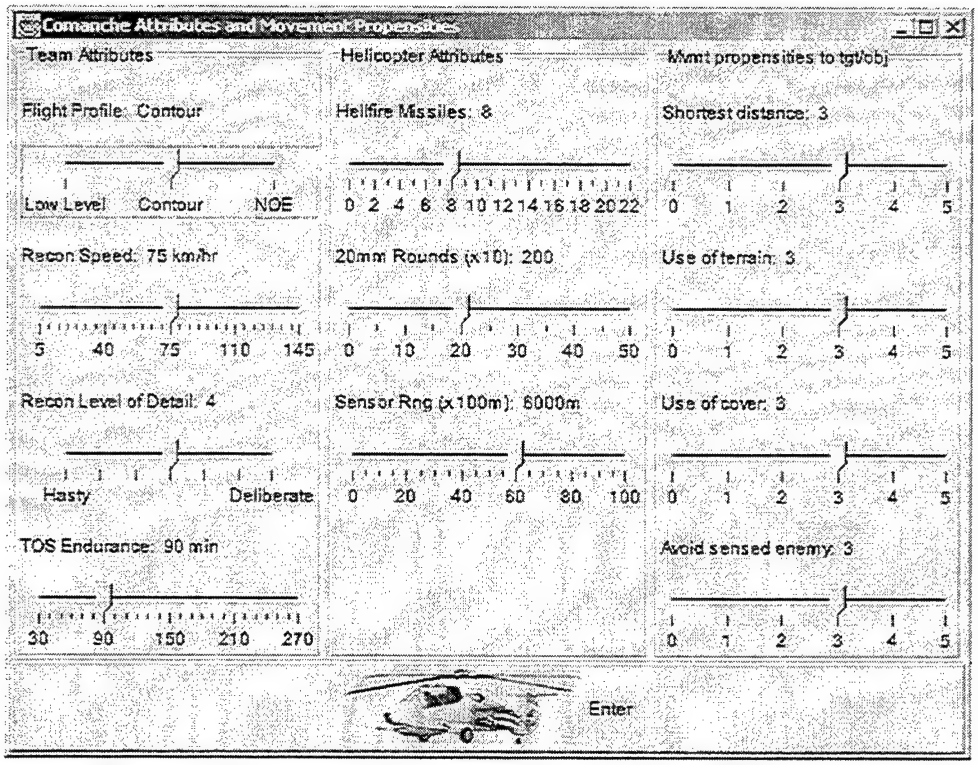


Figure 9. Comanche blue agent instantiation slider panel.

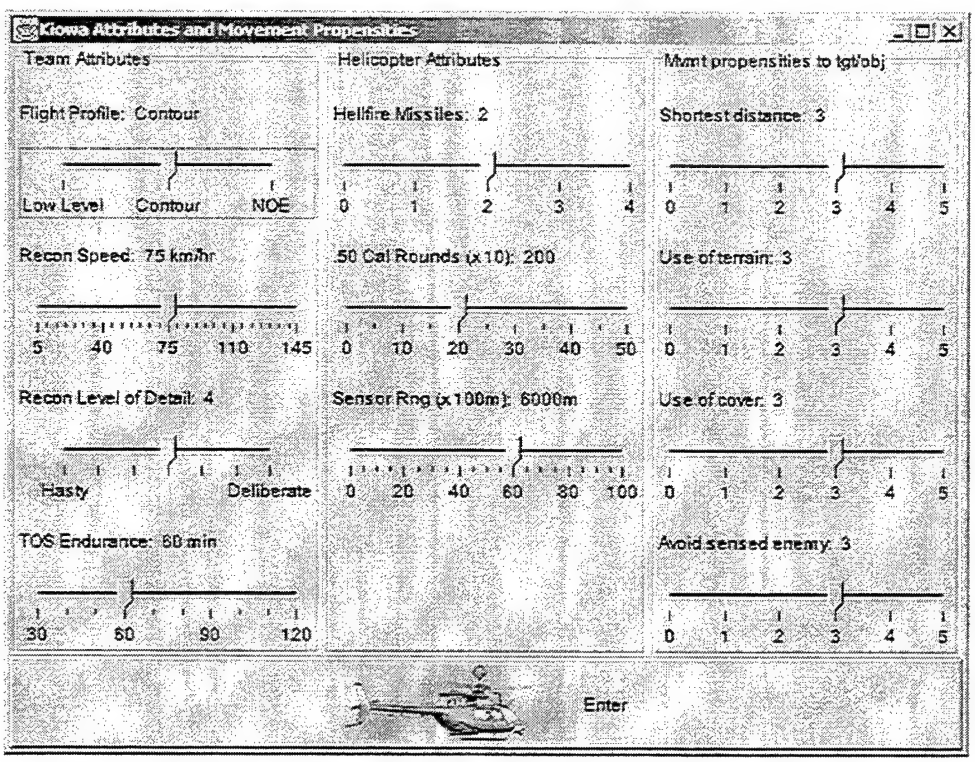


Figure 10. Kiowa Warrior blue agent instantiation slider panel.

The *flight profile* attribute sets the tactical elevation flown by the team while conducting the reconnaissance operation. These choices include: low level, contour, and nap of the earth (NOE). These profiles are consistent and in accordance with U.S. Army Aviation tactical movement methods [TC 1-214, 1992]. The specific implementation of these profiles is covered in more detail in the agent map reading and navigation section.

Blue agent *speed* and *TOS* are implemented in the same manner as explained above for red agents, but they do differ in their ranges. Blue agent slider values for blue agents range from five to 145 KPH for speed. Comanche type blue agent TOS ranges from 30 to 270 minutes (one-half to four and one-half hours), and for Kiowa Warrior type blue agents TOS ranges from 30 to 120 minutes (one-half to two hours). These ranges were set to realistically represent the capabilities of these platform types.

Note that TOS applies only to the actual time within the reconnaissance sector. This time is not inclusive of the time required to travel between an assembly area beyond the Line Of Departure (LOD) (left boundary of the environment screen). This distance varies as dictated by the tactical situation and unit positioning. Therefore, the user must take this time into account when establishing the total TOS for the blue team within the sector.

The *reconnaissance level of detail* team attribute allows the user to set a subjective value to represent the agent's reconnaissance thoroughness. The value of this attribute ranges from one to seven, with one being the hastiest coverage and seven being the most deliberate coverage of the sector. This value is implemented along with the sensor range of the helicopter to set the reconnaissance route checkpoints of navigation

during the movement through the sector. For example, a team with a value of one for reconnaissance level of detail and a 10,000 meter maximum sensor detection range has extensive distance between checkpoints during reconnaissance movement. This implements the agent's desire to conduct a very quick sweep of the zone. The distance between checkpoints is diminished when the reconnaissance coverage level of detail is increased for a blue team.

There are two weapon attributes for blue agents that enable a user to set the *number of missiles* (Hellfires) and *gun rounds*. The slider values for these quantities range the realistic weapon loads for both types of aircraft. Comanche type blue agent missile quantities range from zero to 22 missiles, and for Kiowa Warrior type blue agents missile quantities range from zero to four missiles [OH-58D Operators Manual, 1992 & Crouch, 2000].

The *sensor range* for blue agents operates in the same manner as previously discussed for red agents. Additionally, sensor range is integrated with reconnaissance level of detail (as previously discussed) when setting the teams reconnaissance checkpoints. Additional use of this attribute is covered in more specific detail in the agent scanning and target detection section.

3. Agent Tactical Movement Propensities

There are a total of four agent propensities set by the user that establish an agent's movement tendencies toward the current objective. Agent tactical movement propensities are implemented identically for both red and blue agent types. These propensities were created in much the same manner as the ISAAC agent-based combat model [Ilachinski, 1997].

The first movement propensity is the agent's desire to move towards its current navigation objective in a manner that results in the least distance traveled. The second propensity is the agent's desire to move toward its objective over terrain with the lowest elevation. The third propensity is the agent's desire to move along a route that provides the most vegetated cover and concealment. And the fourth propensity is the agent's desire to move in a direction away from detected enemy.

All four of these propensities range in strength in subjective integer values from one to five. One is the lowest desire, and five is the strongest desire toward a particular movement strategy. The many different combinations of these propensities results in 625 different movement personalities that an agent could possess when tactically moving toward an objective within the sector. How these values are mathematically implemented into the agent's movement behavior is explained in more detail in the agent map reading and navigation section.

4. Agent Logistical Considerations

During simulation movement steps, red agents continuously monitor their status. Red agents have global knowledge of the locations of red re-supply cache sites within the sector. While moving toward their objective, red agents determine whether they can continue toward their final objective, or require re-supply in order to get there. These logistical considerations consist of fuel (TOS), ammunition, or maintenance requirements if hit. In the event that any one of these criteria are met by a red agent during movement, the agent sets its immediate objective to the nearest re-supply cache site along its path. Upon reaching a cache site, red agents remain there for 15 minutes of simulation time and

remain stationary while receiving service. This is a static variable that can be modified in the program code.

Red agent re-supply cache sites are additional objects that must be placed inside the map during environment creation. The cache sites are positioned in the same manner previously discussed for placing terrain features and red agents (see Figure 5). Red re-supply cache sites are represented with red crosses on the map. If a red agent is unable to reach a cache site prior to exhausting its TOS, the agent is stopped, and turns orange indicating its Partially Mission Capable (PMC) status. Note also that blue agents are unable to detect red cache sites, but they are able to detect red agents receiving service at these sites.

In contrast to red agents, there are no re-supply caches or FARPs for the blue agent teams. But, they do possess current knowledge of their system status. When a blue agent's status reaches a critical state, it calls for ROS by another blue agent team. Blue agent ROS criteria include: 20 minutes of remaining TOS, a damaged or killed agent within the team, or ammunition quantities that fall below two missiles and 100 rounds. This determination is made after each simulation event-step by the blue team calling the *checkForROS* method within the *BlueAgentObject* class. If any of these criteria are met, a new blue agent team is instantiated. The newly created relieving team then appears at the LOD and begins movement toward the blue team requesting ROS. The new blue team is instantiated with the identical attributes and propensities initially set for blue agent teams during scenario development. Newly created blue teams conducting ROS do not experience any delay upon their placement into the environment.

In addition to the ROS check, blue agent teams make a critical status check to determine when they should leave the sector for home. Blue team critical states include: 10 minutes of remaining TOS, a damaged or killed agent within the team, or ammunition quantities that fall below one missile and 100 rounds. This determination is made following the ROS check by the blue team calling the *checkForFARP* method within the *BlueAgentObject* class. If any of these criteria are met, the blue team immediately heads for home and enters the enroute profile.

Blue agent teams conducting ROS or returning to home station after relief assume an enroute movement profile. This profile consists of 185 KPH (100 KTHR) and contour flight level. Blue agents conducting the ROS move toward the established ROS location (rally point) set by the blue agent team requesting the ROS at the time of relief. Note that red agents are assumed to be unable to detect or engage blue teams in an enroute profile during a ROS. Additionally, the number of blue agent teams available for requesting ROS is unlimited. If a blue team is requested, it will be instantiated and sent to the ROS rally point location. This is an unreality that can be modified through future enhancements.

E. MAP READING, NAVIGATION, AND MOVEMENT

One of the most difficult and challenging model development issues has been to model the "intelligent" movement of agents throughout the sector of operations. Humans read maps and make cognitive decisions about paths to take when navigating through terrain. The implementation of this concept into agents that move tactically correct, yet not in a scripted and predictable fashion, is very difficult.

Given the grid square and map information described earlier, it has been necessary to specify the way in which agents would read and sense information to intelligently navigate toward their goal. The process used to formulate these decisions can be quite subjective, and has been extensively studied [Stine, 2000]. The intent of this thesis is not to delve deeply into the variety of cognitive aspects of navigation; rather it is to implement an agent map reading and navigation capability that plausibly and generically mirrors human map reading methodology. A route-finding algorithm based on the tactical movement propensities of the agent was devised to achieve this navigation capability.

In order to represent the way humans read maps, agents require access to the map information. Just as pilots and soldiers read the contour lines and vegetated areas on a map, agents are able to "see" where terrain features and cover exist via their access to the map information. With this information the agent must make a decision on the route it will take to reach its goal. This decision depends wholly on the agent's movement propensities, or desired tactical movement profile. The red and blue agent propensities to move via the shortest distance, lowest terrain, over/through cover, and avoid detected enemy agents solely drive an agent's movement within the sector toward its ultimate goal, or current navigation objective.

F. MOVEMENT BEHAVIOR MODELING

Within the instantiation of each agent is the creation of another object called *NextMove*. This class creates an object with reference to the agent's next move on the map. The user sets the movement propensities (as previously discussed) during scenario

development. These propensities are implemented into the agent's movement behavior during the execution of each movement step.

The *MvmtTimer* class handles agent movement turns. Within this class agents are assigned their number of movement credits based on their speed. Agent speeds are set in multiples of 5 KPH, therefore their movement credits equal their speed divided by five. For example, an agent moving 40KPH has eight movement credits per simulation event-step.

An event-step is defined as the complete iteration of a *while* loop (located inside the *MvmtTimer* class) that randomly picks an agent to move next and then executes a single sensor scan and map square movement (one movement credit) for that agent. This *while* loop exits when all agents have exhausted their total number of movement credits available for that event-step.

The random selection of agent movement turns (both blue and red agents) does not allow any one agent the advantage of always getting to move last and possibly sense the previous moves of other agents. Note that if more than one blue agent team is within sector, the teams must alternate turns. Once all agent movement credits are exhausted, agent statuses are reset and a new event-step is initiated. Additionally note that agent engagement determination and execution is a discrete event that happens automatically upon an agent's successful detection of an enemy agent during its scan. See paragraphs G and H of this chapter for a more explicit description of detection and shooting.

1. Movement Algorithm

The movement algorithm involves five primary methods: Two separate methods for red and blue agents for determining the next move, a method for determining bearing

of movement, a method to calculate distances, and a method to calculate the tactical value for a route. Together, these methods formulate the best next map square for an agent to move to in accordance with its movement propensities.

The reason for the two different methods to determine the next move for red and blue agents is the result of the simulated third dimension of the blue agent helicopters. To understand this, envision the blue agents operating from inside the middle of a Rubik's Cube, and red agents moving on top of a checkerboard. From the middle cube of the Rubik's Cube, blue agents are able to move to 26 surrounding cubes, whereas red agents are only able to move to eight surrounding, two-dimensional squares. Although each red agent's horizontal movement is two-dimensional, their respective three-dimensional elevation on the map is maintained for their present map square location.

The agent movement determination process is generally the same for red and blue agents, except that blue agents account for altitude. The movement algorithm begins by selecting the appropriate *determineNextMove* method for the agent whose turn it is to move. Within this method it is initially checked to see if the agent is at its current navigation goal. This can be just a navigational checkpoint, a re-supply cache, or the agent's ultimate objective. If it is determined that the agent is not at a navigation objective and currently does not have an objective locked into his navigation route, a new objective must be determined. The agent begins this sequence shooting a direct bearing to its main objective. The algorithm then iterates through the grid locations along that bearing toward the main objective. Elevation, vegetation coverage, and distance information are collected at each grid square along the bearing toward the objective. The location is also checked to see if it is the same as the objective and whether there has

been a change in elevation with respect to the agent's current location. A sensor scan prior to the move tells the agent of any sensed enemy locations within its sensor range. This result is additionally added into the agent's overall determination of the next location to which to move.

2. Alternate Route Finding

If a change in elevation from that of the agent's current location is noted during the traversal along the bearing to the main objective, the grid location of the point of elevation change is saved and a Boolean flag is tripped requiring that alternate paths to the objective be checked. The check of alternative routes determines whether there is a better path to the objective that will better support the agent's tactical movement propensities. This checking of alternate routes is implemented through the route finding algorithm.

At this point, determination of the next move enters a loop that checks alternate paths via deflection points 20 map squares orthogonal to the point of the elevation change, and for all four cardinal directions from the agent's location (see Figure 11). If at any time the alternate route penetrates the boundary of the environment, the loop for that cardinal direction is broken and the alternate route in that cardinal direction is no longer a viable option. All four cardinal directions must be checked for alternate routes due to the unpredictable navigation bearing of the agent. Following the traversal of an alternate route, the collected route values (distance, terrain, cover, and enemy) are evaluated for total tactical value calculation and comparison. Additionally note that no two alternate routes are ever the same or repeatedly checked.

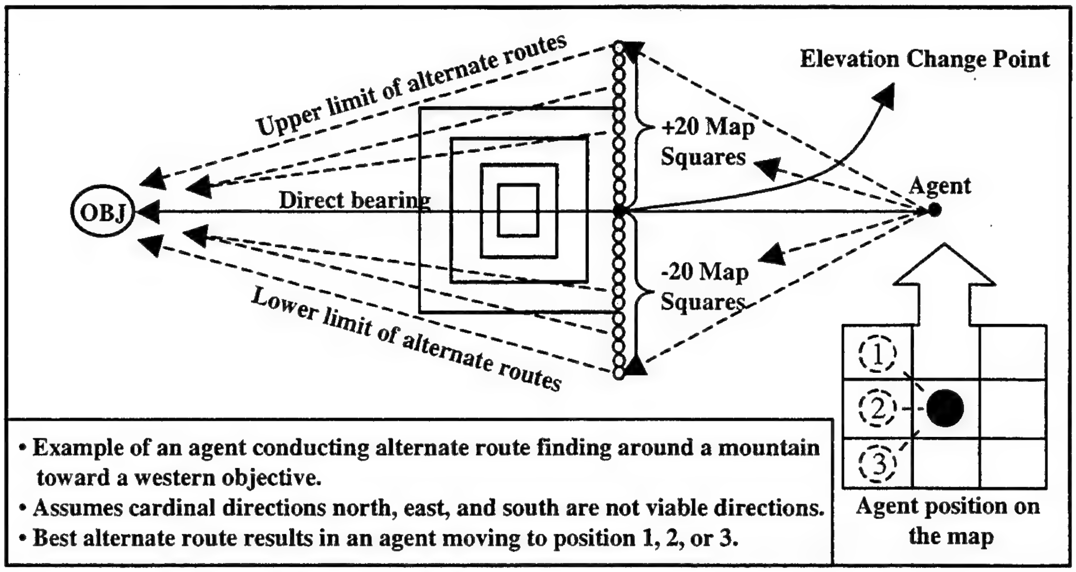


Figure 11. Alternate route finding.

3. Route Tactical Value Calculation

When a route is successfully checked to its objective, the values collected for distance, elevation, cover, and sensed enemy are sent to the method for determining the tactical value of that route with respect to the movement desires of that agent. This method multiplies the appropriate values by the percentage of importance (propensity factor) given to that propensity. These four values are then added to give an overall tactical value for moving along that proposed route. See Figure 12 for a depiction of the tactical value calculations. The route returned with the highest tactical value results in the next map square move (new location) for that agent being set to the first grid square along the bearing of that route.

4. Integration of U.S. Army Aviation Tactical Flight Profiles

With the three-dimensional capability of the blue agents comes the integration of the flight profile previously described for U.S. Army Aviation tactical movement. Army

Aviation teaches three methods for conducting terrain flight during tactical operations; they are: NOE, contour, and low-level. NOE is defined as flying as close to the earth's surface as safely possible. The speed of the aircraft varies and altitude varies up to 50 feet above the terrain. Contour flight still conforms to the contours of the earth, but at a higher altitude. Aircraft speed still varies, but since a higher altitude is maintained, it requires fewer altitude changes to accommodate to changes in the terrain elevation. The aircraft is maintained in trimmed flight, and altitude varies between 50 and 200 feet above the terrain. Low-level flight resembles straight and level flight. It sets a profile that requires the least fluctuation in aircraft speed and altitude. The intent is to enable faster movement with smaller power and pitch adjustments. When choosing one of the profiles discussed above, the primary tactical concern is: what type of enemy template is expected, and what type of terrain is encountered [TC 1-214, 1992].

These three profiles were then integrated with the elevation dimension to enable a blue agent to move through the sector at different altitudes during the simulation. If NOE is chosen, the blue agent desires to fly at an elevation level of two. Since it costs an agent more movement credits to climb over higher elevation, an agent flying NOE is apt to follow the lowest terrain during reconnaissance. If contour is chosen, the agent desires to fly at an elevation level of three. With this type of profile, an agent is required to make fewer power adjustments when navigating toward an objective. Since the agent flies at the third elevation level, it is not required to increase its altitude for elevation levels one and two, thus resulting in a straighter route that is habitual to contour flight. Finally, if low-level is chosen, the agent desires to fly at an elevation level of four. This profile

results in an altitude that rarely requires increase because terrain elevations of one through three all fall under the agent's desired movement altitude.

Tactical Value for Route= $d_{TV} + tr_{TV} + c_{TV} + e_{TV}$

<u>Adjusted Propensity Factors</u>

d_{PF} = (shortest distance slider bar value / slider bar total)
tr_{PF} = (use of terrain slider bar value / slider bar total)
c_{PF} = (use of cover slider bar value / slider bar total)
e_{PF} = (avoid sensed enemy slider bar value / slider bar total)

<u>Propensity Tactical Values</u>

$$d_{TV} = d_{PF} \bullet [\Delta\, d_{agent}]$$
$$tr_{TV} = tr_{PF} \bullet [\Delta\, mvmt\ Cost]$$
$$c_{TV} = c_{PF} \bullet [\Delta\, cover]$$
$$e_{TV} = e_{PF} \bullet [\Delta\, d_{enemy}]$$

The parameters have the following meaning (measured in map squares):

- $\Delta\, d_{agent}$ = (direct-bearing rte distance to objective - alternate rte distance to objective)
- $\Delta\, mvmt\ Cost$ = (direct-bearing rte mvmt cost to objective - alternate rte mvmt cost to objective)
- $\Delta\, cover$ = (direct-bearing rte cover value to objective - alternate rte cover value to objective)
- $\Delta\, d_{enemy}$ = (direct-bearing rte distance from enemy - alternate rte distance from enemy)

Figure 12. Route tactical value calculations.

All three of these profiles have very distinct differences that add many interesting factors to the results of simulation runs. For example, low-level flight entails covering less terrain, therefore covers an area more quickly. But, NOE makes better use of the terrain as cover, thus enhancing survivability. Higher altitudes offer better visibility, but also enhance the chances of red agents detecting the blue agents. The impact of these profiles is clearly demonstrated during simulation execution and is discussed in greater detail in Chapter 4. See Figure 13 for a graphical depiction of these three flight profiles.

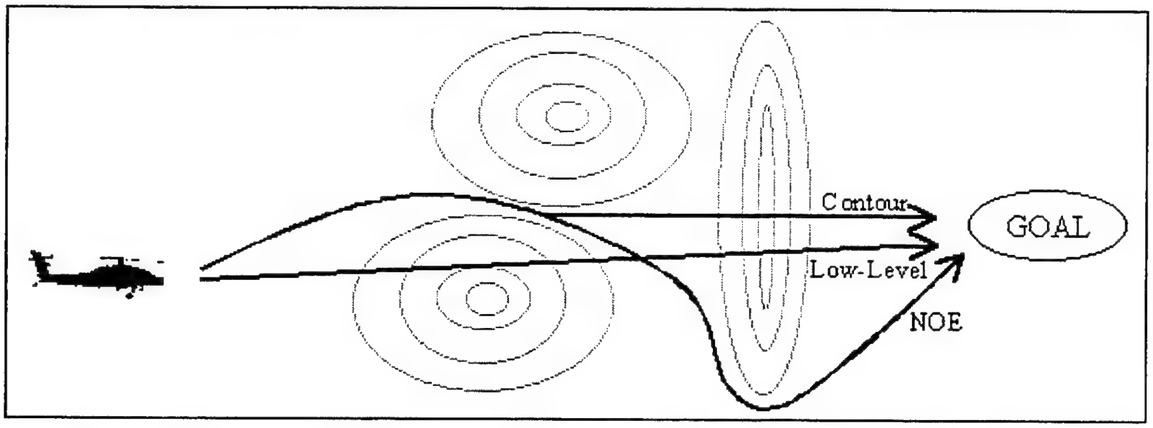


Figure 13. U.S Army Aviation tactical flight profiles.

Additionally, a blue "vapor trail" enhancement was implemented to visually depict the path taken by blue agent teams during reconnaissance of the sector. This enables the user to visually observe areas reconnoitered and areas of significant situation development. It also visually emphasizes the routing impact created by the three different types of tactical profiles. Note that the vapor trail does not appear behind blue agent teams conducting ROS.

G. AGENT SCANNING AND TARGET DETECTION

The author of this thesis independently developed the detection algorithm used in this model. As is true of navigation, the modeling of target detection is abundant, and varies in its findings concerning how people visually detect, acquire, recognize, and identify possible targets. The intent of this thesis is not to delve deeply into the many different areas that this model touches upon. What is needed is a temporary detection algorithm that makes logical sense, incorporates the information sensed by the agents, and integrates with the systems and parameters utilized in this model. It should be noted that no implementation of fratricide was put into this program, and a successful detection is always considered successfully recognized and identified. This is an unreality that can be modified through future enhancements.

The algorithm used for agent scanning and detection of opponent agents is very methodical and comprehensive. In order to enhance the efficiency of the program's code, a reverse approach was used when conducting scans. Instead of iterating through the entire map, probabilities of detections are checked only after it is determined that an enemy agent is within sensor range and Line Of Sight (LOS) of the searching agent. This method of determining detection tremendously enhances the program's execution efficiency.

As previously stated, agents already possess knowledge of the map for navigation purposes. This navigation knowledge only provides terrain data, not locations and actions of opponent agents. Therefore, agents need a different method of conducting limited area searches associated with their sensor systems. Sensor ranges, terrain, speed, and bearing all play major roles in determining a successful target acquisition.

Prior to each agent's next move (previously discussed), the agent conducts a scan of his surrounding area for opponent agents. This scan is initiated by calling the appropriate scan method in the *NextScan* Java class. When each agent is instantiated, a *NextScan* object is established enabling the agent to have access to the *scan*, *detect*, and *shoot* methods. For example, if the next agent to move is a red agent, the *scanForBlueAgents* method is called to begin the target acquisition algorithm.

The sensor scan works much like the traditional "cookie-cutter method". The first step sets the agent's sensor area based on the agent's sensor range. For example, an agent with a sensor range of 5,000 meters is able to sense an area 1600 map squares around him (given 360 degrees sensibility and the 250 by 250 meter map squares previously discussed). Once this sensor range is established, the searchable area within the map is established. Note that this search area is a square area.

Next, nested *for* loops are used to iterate through this search area, and step through the various detection checks of the algorithm. These *for* loops start the grid square checks in the upper left hand corner of the agent's searchable area and continue down the columns of grid squares within the searchable area until the agent's entire sensor area is checked. These progressive checks implement the code efficiency mentioned above.

The first detection check determines whether the map square location being scanned is within the boundaries of the map. If the location is not a viable square to check, there is no need to continue along the algorithm, therefore the loop is incremented to the next square for scanning. If the location *is* a viable grid square, the square is checked for its occupation status. Since the algorithm only cares if there is actually an

agent located in the grid square, unoccupied squares stop the algorithm and increment the loop onto the next grid square. This checking of occupation really increases the program efficiency since the majority of grid locations are not occupied. In the event that the grid location is occupied, it must be determined if the occupation is an opponent agent. Obviously this means for red agents that the occupation must be a blue agent, and for blue agents, it must be a red agent.

If the algorithm has been successful thus far for the map square being checked, the map square in question must possess an opponent agent. At this point in the algorithm, a LOS check to the opponent agent must be made. Here the elevation levels and tactical flight profiles previously discussed become factors. Obviously, an opponent agent cannot be sensed if it is not within a clear LOS from the detecting agent. Therefore, a new method for determining the LOS was created. For example, if the searching agent is a red agent, the method it calls is named *determineLOStoBlueAgent*.

The methods for determining the LOSs begin by setting the location of the searching agent and target agent, setting the target bearing from the searching agent to the target agent, and setting the slope from the searching agent to the target agent. Once these variables are established, a *while* loop is entered to traverse along the bearing toward the target and check for higher terrain elevations that block the LOS from the searching agent to the target agent. If the slope from one of the grid location elevations being checked along the bearing exceeds the initial slope established between the searching and target agents, the target agent *is not* within a clear LOS and a false Boolean value (undetectable) is returned to the scan method. If the target agent *is* successfully reached while traversing along the bearing and the initial slope was not exceeded, the

target agent *is* within the LOS of the searching agent and a probability of detection must be determined next.

The final detection calculation incorporates five variables: distance to the target, bearing to the target, speed of the target, speed of the searching agent, and cover associated with the location of the target agent. The first four variables are retrieved from within the program and then normalized with respect to the agent's sensor range. The probability of detecting the target agent increases as the distance to the target decreases, bearing offset from the target decreases, speed of the target increases, or speed of the searching agent decreases. See Figure 14 for depiction of the formula and parameter explanations.

It was assumed that cover and bearing to the target play a more significant role than the other variables in determining success of detecting a target. Therefore, once the probability of detection was obtained, as explained above, the cover status of target agent's location and the magnitude of bearing offset from the target are checked. If the target agent is determined to be under cover and concealment (depicted by green on the map), the probability of detection is reduced by 75%. If that target's bearing from the searching agent was greater than 45 degrees, detection was reduced by another 50%. Additionally, target agents outside of the 180 degree fan from the scouting agent's movement bearing are not detectable. These are subjective values implemented by the author. These values can be adjusted by changing the variable values in the program's code.

$$\mathbf{Pr(D)} = \frac{[(\alpha \bullet f_B + (T\upsilon \bullet f_{T\upsilon}) + (S\upsilon_{max} \bullet f_{S\upsilon}) + \Delta_R]}{(4 \bullet S_{SR})}$$

The parameters have the following meaning:

- Pr(D) = Probability of Detection / movement step
- α = (90° - scout bearing offset from target)
- f_B = (bearing adjustment factor) $\Rightarrow$ (sensor range / 90°)
- $T\upsilon$ = (target speed)
- $f_{T\upsilon}$ = (target speed adjustment factor) $\Rightarrow$ (sensor range / max capable speed of target agent)
- $S\upsilon_{max}$ = (max capable speed of the scout – actual scout speed)
- $f_{S\upsilon}$ = (scout speed adjustment factor) $\Rightarrow$ (sensor range / max capable speed of scout agent)
- Δ_R = (scout sensor range – target range)
- S_{SR} = (scout sensor range)
- note: 4 is used to convert 250m x 250m map squares to kilometers

Figure 14. Agent detection algorithm calculation.

Finally, a uniform random number draw resulting in a value less than the final calculated probability of detection results in a successful target agent detection. At this point the hunting agent becomes a killer and must deal with the target agent accordingly.

Blue agents exhibit an additional investigative search behavior given the successful detection of a red agent. This intelligent behavior is used to represent a blue agent's desire to further reconnoiter and develop the situation in an area containing red

agents. This investigative search is incorporated through the establishment of a predetermined search path. Upon a successful detection, the blue agent adds four additional navigation checkpoints that are three kilometers in all four cardinal directions from the location of the detected red agent. The blue agent immediately changes its bearing of navigation to begin reconnaissance along this investigative search-path. If the blue agent is north of the enemy location, the search-path is traversed in a counter-clockwise direction. If the blue agent is to the south of the enemy location the search-path is traversed in a clockwise direction. Note that if another red agent is detected during an investigate search pattern, new search-path navigation checkpoints are immediately added to its path. This implementation adds a type of pursuit behavior to the blue agent's desire to find enemy red agents.

H. AGENT ENGAGEMENTS AND SHOOTING

The area of tactical engagements, probability of hit (P(H)), and the conditional probability of kill given a hit (P(K|H)) are other quantities to which there are many different approaches throughout the military community. There is often debate on the effectiveness of certain weapon systems against various enemy weapon systems, given different conditions. Additionally, the data concerning most military weapon systems and ammunition is usually classified. Consequently, unrealistic or unfamiliar data may cause critics to question the accuracy of any type of model. The data used for this model was obtained from the U.S. Army's combined arms simulation trainer, JANUS, version 7.06D [JANUS, 1999]. The reader should keep in mind that these P(H) and P(K|H) data serve as placeholders subject to future modification. Alternative data sources can be

applied. See Appendix A for depiction of the applicable P(H) and P(K) tables obtained from JANUS.

The JANUS data are not comprehensive, nor are they perfectly integrated with the weapon parameters established for this model, but they do provide needed preliminary information. The primary areas in which JANUS does not contain the desired data are for the new Comanche and various other helicopter weapons systems used on U.S. Army helicopters. Since the Comanche is not yet fielded, data specifically related to its platform, when in actual operation, are not available. But, given that many of the weapons are similar to those found on the AH-64 Apache (Hellfire missiles and 30mm gun), that data were used without adjustment to represent the Comanche and Kiowa Warrior weaponry. Other assumptions include no available data on the .50 Caliber gun for the OH-58 Kiowa Warrior, and no available data for rockets. To adjust for such missing data, rockets were not included, and Hellfire and 30mm data for the Apache were implemented for the Comanche 20mm and Kiowa Warrior .50 Caliber guns.

For red agent P(H) and P(K|H) data, JANUS data were solely used. Only the primary weapons system for red agents was implemented in this model. Red agent primary weapon system's P(H) and P(K|H) against the Apache were implemented for the Comanche. Kiowa Warrior data were directly obtained from JANUS.

The engagement algorithm uses a discrete, uniform, randomized draw once a target is detected to determine whether a target is hit or killed. There are separate *determineHitOrKill* Java methods for red and blue agents. These methods begin by establishing the weapon ranges, resetting the Boolean variables, and determining the range to the detected target agent. Once information is set it is determined whether the

target agent is within range of the shooting agent's weapon. For blue agents, there are additional checks for choosing the "best" weapon for engaging the target. Recall that blue agents possess two different types of weapons, Hellfire missiles and a gun (primary and secondary). Blue agent weapon systems have different ranges and effects. For example, gun rounds are ineffective against armored vehicles like the T-80 tank. For red agents, there is only one weapon system; therefore the range to the target is the only limiting factor when determining whether a target agent can be engaged. Red agents do not have weapon choices or different target effectiveness criteria.

Once the weapon system is chosen, various checks are made to determine the P(H) and P(K|H). To obtain these values, JANUS factors in whether the target is moving, the shooter is moving, and if the target is over/under cover. After choosing the appropriate conditions, the P(H) and P(K|H) are determined by making a calculation that interpolates between the range to the target and the maximum effective range of the engaging weapon. Once the P(H) and P(K|H) are established, it is compared with the previously drawn uniform random numbers for P(H) and P(K) values. If both random values are less than the P(H) = P(K|H) for that weapon system and condition, the target agent is killed. If the random P(H) value is only less than the weapon system P(H), the target agent is injured (hit). If neither value is less than the P(H) = P(K|H), the shot is considered a miss. Note that the determined P(H) and P(K|H) values for a weapon system are identical. The only difference between a hit and kill is that it requires two separate random comparisons. See Figure 15 for a symbolic representation of the simulation of a target being hit and a target being killed. Additionally, once a weapons system is fired, its remaining rounds are reduced accordingly.

Randomly drawn (D) probabilities in program	$P(H)_D = \text{uniform}(0,1)$ $P(K)_D = \text{uniform}(0,1)$
Probabilities for applicable weapon (W) system & conditions (See Appendix A)	$P(H)_W \;= P(K\|H)_W$
Adjudication of hits & kills in model	$P(H)_D > P(H)_W \Rightarrow$ **miss** $P(H)_D < P(H)_W \Rightarrow$ **hit** $P(K)_D < P(K\|H)_W \Rightarrow$ **kill**

Figure 15. Probability of hit and kill formula.

The completed engagement adjudication results in the appropriate modifications to the shooting and target agents. Injured or hit red agent movement credits are reduced (by half), their color is changed to orange, and their new navigation objective becomes the nearest re-supply cache site. Blue agent teams that incur a hit turn orange and immediately return home while calling for ROS. Killed red agents are stopped, turn black, and remain visible on the map, but no longer participate in the simulation. A blue cross is used to identify a killed helicopter within a blue agent team at the location killed. Note that injured agents (orange) are still susceptible to being engaged and killed. Upon completion of this algorithm (if required), control of the simulation is returned to the *MvmtTimer* Java class iterating the simulation event-steps for execution of the next set of agent movements.

I. MODEL EXECUTION

Two Java classes named *ReconSim* and *MvmtTimer* handle the overall execution and running of the model. The *ReconSim* class actually instantiates the simulation and calls the *MvmtTimer* class to initiate and manage the event-step iterations of the model execution. *ReconSim* opens a blank environment that enables the user to create an environment, map, and scenario. It also instantiates a reference to the *MvmtTimer* object which remains idle until the simulation is started by the user. Once the user creates an entire scenario and chooses to run the simulation, the environment is saved into a temporary environment file. This enables the simulation to replicate the scenario several times, if so desired by the user. Upon pressing the *Start Sim* button by the user, the *MvmtTimer* object is instantiated awaiting the simulation parameters setting by the user.

It is important to note that a new environment and a blue agent team must be created in order for the simulation to run. By creating a *new* environment (or *load*ing a previously saved environment) a *Map* object is instantiated, giving the simulation a template to operate upon. If the user tries to run a simulation without creating or loading an environment first, the program will not properly execute and exception errors will occur. Environments may be saved with any combination of objects and agents, but must have a blue team instantiated prior to execution. If a simulation is started without an instantiated blue team, the simulation will prematurely stop, thinking the simulation is complete due to no blue teams remaining in the simulation.

Once the user creates a complete environment and scenario, the simulation is ready for execution. The user chooses to run the simulation by pressing the *Start Sim* button at the bottom of the environment window (see Figure 4). By selecting the *Start*

Sim button, the user is presented with the *Simulation Parameters* window that allows the user to set the execution parameters for running the simulation (see Figure 16).

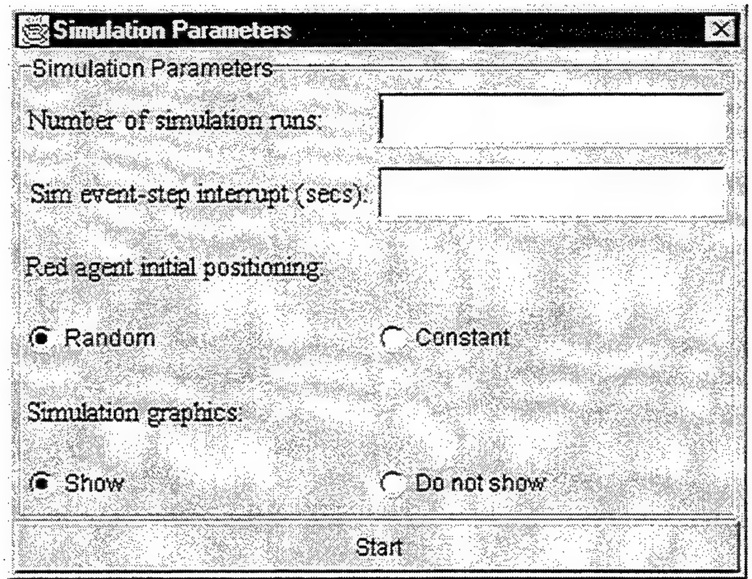


Figure 16. Simulation parameters dialog box.

The *Simulation Parameters* window gives the user four options for simulation execution. Those four options include: number of simulation runs, time between simulation event-steps, red agent initial starting locations, and showing of simulation graphics. The user must input the time and number of simulation runs prior to execution, but red agent positioning and showing of graphics have default parameters of *Random* and S*how* respectfully.

The *Number of simulation runs* option allows the user to input the number of replications desired for executing the current scenario. The input for this option is any integer value greater than zero. The *Math.Random* java class is used to generate all random numbers in this program. Therefore, the random numbers used in each simulation run are always different. This java class generates random numbers using the

computer's clock. This results in the "true" agent movement randomness that was desired in this model's development.

The *Sim event-step interrupt (secs)* option allows the user to set an additional artificial pause time between the simulation's calls to the *MvmtTimer* object. This option merely controls the speed of the simulation's visual display. The input for this option is any positive real number (including zero) in decimal or whole integer format representing seconds. A simulation interrupt-time of zero results in the simulation running as fast as the computer's processor can execute the calculations and render the graphics (if *Show* is selected). Times from zero to two seconds are common inputs for this option to optimally view the simulation. Given that agents must move at a minimum speed of 5KPH and map squares represent 250 by 250 meters, a simulation interrupt-time of 180 seconds represents viewing in "real-time".

The *Red agent initial positioning* option allows the user to have initially-created red agents start in the same location (*Constant*) for replicated simulation runs, or be randomly (*Random*) repositioned at the start of each new run. If the *Random* option is selected, red agents will begin each new simulation run from a randomly chosen location within 5 KM of their initially created location. This option introduces an interesting variability in red agent movement behaviors when negotiating the terrain objects created for the simulation scenario. The default position for this option is *Random.*

The *Simulation graphics* option gives the user the option of viewing (*Show*) the simulation's graphical display during execution, or disabling (*Do not show*) the graphical display. If the *Do not show* option is selected, a *Please wait while simulations run* dialog box is displayed while the simulation(s) execute (see Figure 17). Additionally, the

simulation interrupt-time is automatically set to zero for optimal processor performance. The default position for this option is *Show*.

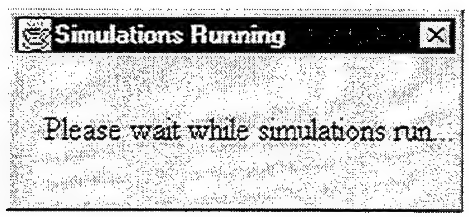


Figure 17. Please wait dialog box.

Once a scenario is properly created and the simulation parameters are set, the user can begin execution of the simulation by pressing the *Start* button. The simulation replication completes once a blue team reaches the last checkpoint at the Limit Of Advance (LOA – far right of the environment window), and returns home across the LD (far left of the environment window) without requiring any more ROSs. At this point no more blue team agents remain in blue agent vector and the simulation run knows it is complete. If more replications are left to run, the environment is reloaded and run as previously discussed. Once all replications are complete, the statistics box is presented to the user as discussed in the next section.

J. SUMMARY STATISTICS GATHERING AND REPORTING

Users are able to pause the simulation and view the current status of an agent at any time. Additionally, results and data are collected during the execution of simulation runs for analysis and viewing upon completion of all simulation iterations. The user has the capability to archive individual simulation-run statistical results within output files, and is presented with an overall statistical summary of all simulation runs upon completion.

1. Current Agent Status

The user is able to pause a running simulation at any time by pressing the *Pause/Continue* button at the bottom of the environment panel (see Figure 4). By pausing the simulation, the user is able to click on an agent and view its current status. When the user clicks on an agent, an agent dialogue box is presented listing the 11 current status conditions of an agent (see Figure 18). These 11 status conditions include:

- Type; blue team, or red and the equipment type
- Current (x,y) location in accordance with the computer screen orientation previously discussed
- Elevation; flight profile for blue agents (NOE, Contour, or Low Level), or ground elevation level for red agents (1-6)
- Maintenance status; FMC, PMC, or NMC (fully operational, injured, or inoperative/dead)
- Remaining TOS; hours left in sector (one-place decimal format) until agent must be relieved or refuel
- Movement speed in KPH
- Number of primary weapon rounds; total missiles for blue teams or total engagements for red agents
- Number of secondary weapon rounds; total rounds for blue teams (N/A for red agents)
- Sensor range in thousands of meters
- Action or situation the agent is currently conducting or in; scouting, engaging, relieving, returning, or stationary
- Agent's current navigation objective (x,y) location in accordance with the computer screen orientation previously discussed

2. Collecting Individual Simulation Run Summary Statistics

Individual simulation run statistics can be output to *out* files prior to the execution of the program. As an example, an output file can be created when beginning the program at the DOS prompt by entering a command:

C:\Simulation> java ReconSim > scenario.out

In this example, *Simulation* is the directory containing the program, *java ReconSim* executes the program, and *scenario.out* is the file created to collect the statistical data for each individual simulation run. Following the completion of simulation run(s), this data output file containing the summary statistics for each run can be imported into a statistical analysis tool such as Microsoft's Excel or S-Plus.

3. Individual Simulation Run Summary Statistics

Eight separate summary statistics are collected within the output file for each simulation run when the above command is entered. The values for the eight statistics are represented to the nearest tenth. Those eight individual summary statistics for each run include:

- Total number of red agents successfully reaching their final objective plus red agents remaining fully operational at run completion
- Total number of red agents initially hit but not killed by blue agent teams
- Total number of red agents killed by blue agent teams (this number includes agents subsequently killed that were previously hit)
- Total number of blue agent teams used to complete reconnaissance of the sector
- Total number of missiles used by blue agent teams during reconnaissance of the sector

- Total number of rounds used by blue agent teams during reconnaissance of the sector

- Total number of blue agent helicopters hit by red agents during reconnaissance of the sector (recall that a blue agent team represents a two-ship helicopter team)

- Total number of blue agent helicopters killed by red agents during reconnaissance of the sector (recall that a blue agent team represents a two-ship helicopter team)

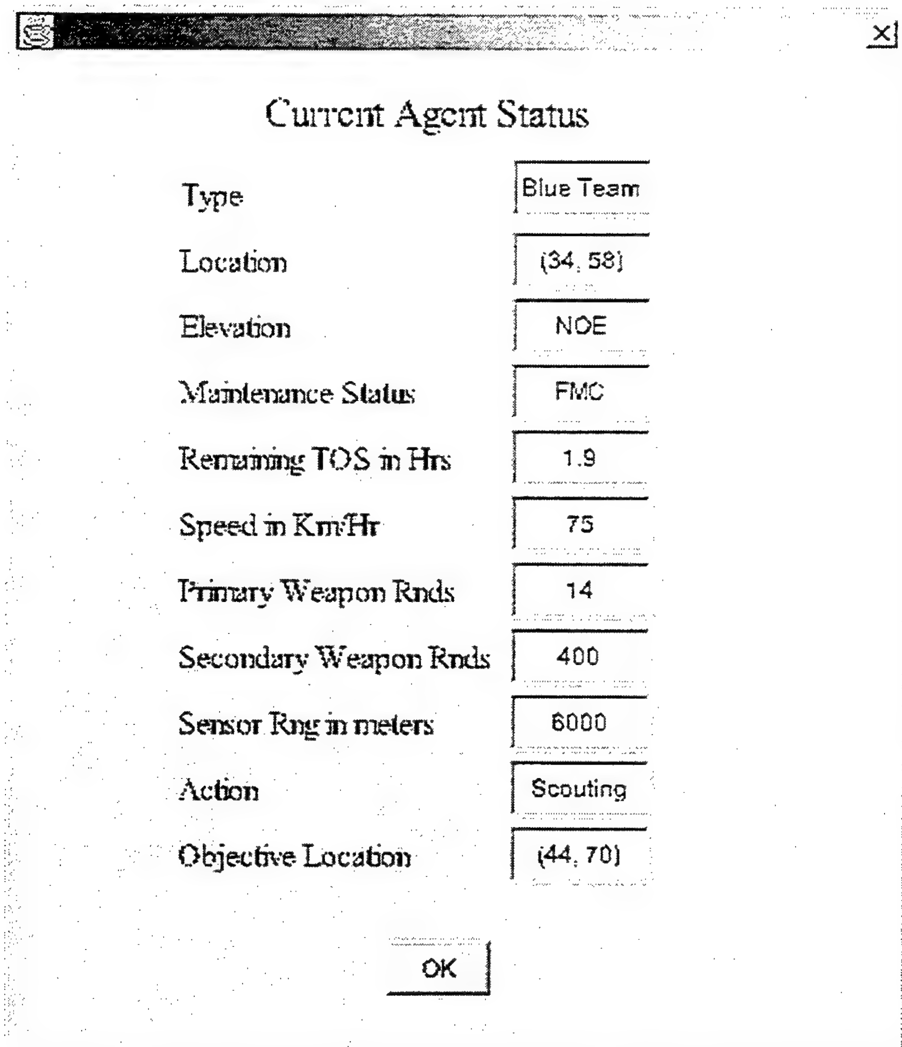


Figure 18. Current agent status dialogue panel.

4. Overall Summary Statistics Reporting

An overall statistical report is presented to the user upon simulation completion, regardless of the user's choice to collect the above statistics for individual simulation runs. This final report is presented to the user via a Java panel with nine statistical values presented in one-place decimal format with percentages when applicable (see Figure 19). The nine overall summary statistics for all runs include:

- The total number of red agents instantiated for each simulation run
- The average number and percent of red agents successfully reaching their final objective plus those remaining fully operational per run
- The average number and percent of red agents initially hit but not killed per run by blue agent teams
- The average number and percent of red agents killed per run by blue agents (this number includes agents subsequently killed that were previously hit)
- The average number of blue agent teams per run used to complete reconnaissance of the sector
- The average number of missiles used per run during reconnaissance of the sector
- The average number of rounds used per run during reconnaissance of the sector
- The average number and percent of blue agent helicopters hit by red agents per run during reconnaissance of the sector (recall that a blue agent team represents a two-ship helicopter team)
- The average number and percent of blue agent helicopters killed by red agents per run during reconnaissance of the sector (recall that a blue agent team represents a two-ship helicopter team)

Once the user is finished viewing the overall summary statistics, the OK button is pressed and the scenario's environment is cleared.

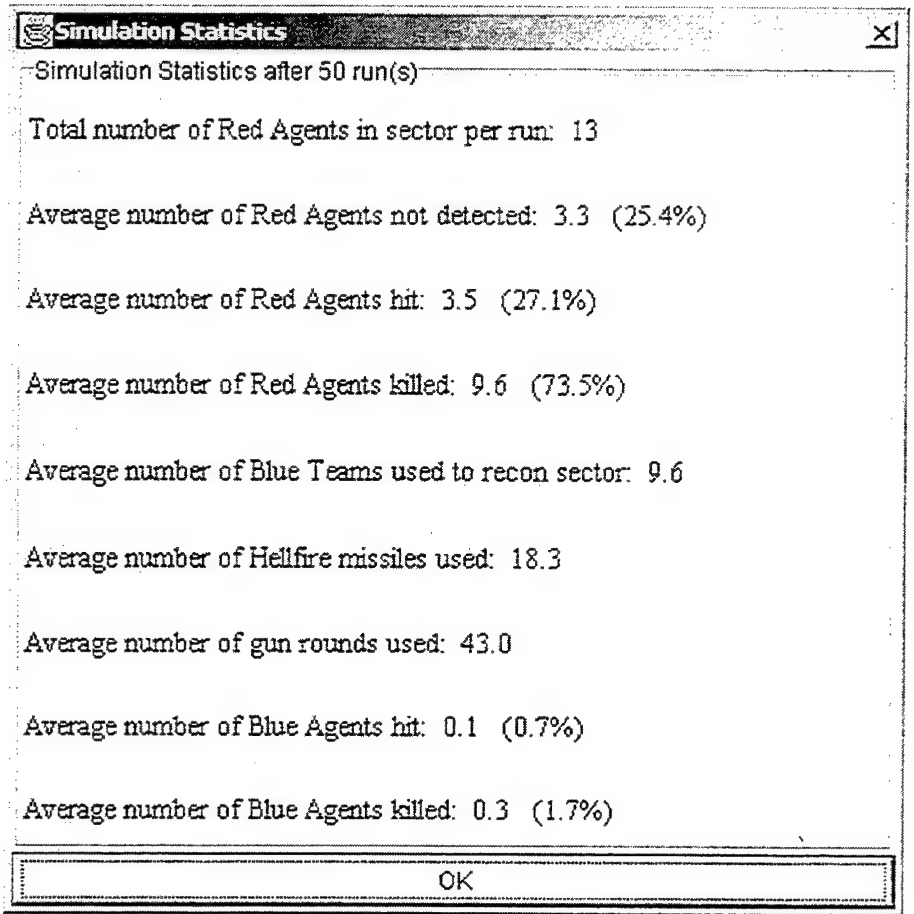


Figure 19. Overall simulation statistics dialogue panel.

IV. MODEL ANALYSIS AND RESULTS

A. INTRODUCTION

The intent of this chapter is to investigate the usefulness and potential of the model by using it to analyze the effect of various blue agent performance parameters against a generically created terrain, and red agent enemy scenario. It is impossible for this thesis to demonstrate all of the capabilities, characteristic combinations, or tactical attributes that this model possesses. The agent characteristics and attribute combinations, along with terrain and enemy template scenarios, are boundless. The scenarios analyzed in this chapter show one method of analysis to explore the performance of tactical profiles, helicopter attributes, and logistical forecasting for blue agent teams against a templated red agent enemy situation and terrain environment.

B. METHOD

This paragraph defines the area of investigation, environment, agent profiles, and experiments used throughout this chapter's analyses.

1. Areas of Investigation

The area of investigation analyzed with the following scenario is the performance of various blue agent helicopter team profiles and characteristics against a realistically represented enemy force, generically termed red. Specifically, model statistics are analyzed to investigate the ability of the model to produce useful and realistic results that are consistent with U.S. Army Aviation tactics and procedures. Once this has been determined, it is of further interest to create a blue agent team that performs with the best combined mission success and survivability with respect to the created environment.

Finally, a logistical synopsis is used to depict the ammunition and asset requirements forecasting capability of the model.

2. Terrain

The terrain created for this scenario utilized all terrain objects available within this model. It consists of arbitrarily positioned mountains, hills, ridges, and valleys. Additionally, vegetation is sporadically positioned on the terrain to represent various forested areas. The terrain is developed with no intent to create advantageous avenues of approach for either agent type; the terrain could be patterned after that of a real geographical region. Red agent re-supply cache sites have been positioned throughout the sector for red agents to use as necessary during their navigation through the terrain. These cache sites are strategically placed within areas of vegetated cover to represent an enemy's intent to conceal their supply points. See Figure 21 for a visual depiction of the particular terrain created for these experiments.

3. Red Agent Enemy Force Representation

The array of red agent enemy forces placed within the environment represents the typical forward reconnaissance assets positioned forward of a Soviet Motorized Rifle Regiment (MRR). The types of red agent vehicles within this model do not encompass all of the different types of vehicles found in these types of enemy units, but they do include the major systems of threat, and hence are of primary interest to helicopters conducting reconnaissance.

The Soviet units commonly deployed forward of a MRR consist of a regimental reconnaissance element, followed by a battalion-size advanced guard [FM 100-2-3, 1991

& FM 100-63, 1996]. The regimental reconnaissance element is deployed forward of all other MRR forces, including the advanced guard, to provide early warning for the regimental main body. The advanced guard is comprised of a platoon-size Combat Reconnaissance Patrol (CRP), followed by a company-size Forward Support Element (FSE), followed by the advanced guard's main body. The CRP and FSE are the advanced guard's primary reconnaissance elements. The CRP is used to provide early warning about enemy strength and composition, while the FSE is used to engage those lead enemy elements. The distances between these elements usually range between five to ten kilometers.

The scenario created for these analyses implemented the regimental reconnaissance element, CRP, and FSE. All five types of red agent vehicles available to this model are used to represent the above described unit elements. For this scenario, the regimental reconnaissance element is comprised of two BRDM wheeled reconnaissance vehicles and one 2S6 air defense vehicle. The CRP is comprised of three T-80 tanks, two 2S6 air defense vehicles, and one BMP armored fighting vehicle. The FSE is comprised of six T-80 tanks, three BMP armored fighting vehicles, and two ZSU-23-4 air defense vehicles. This totals 20 red agent enemy vehicles instantiated for this scenario's enemy unit representation. See Figure 20 for a visual depiction of this enemy composition and array.

4. Red Agent Profiles

Red agent attributes and movement propensities were created to represent realistic equipment performance and tactical propensities. Tables 2 and 3 depict the attributes and propensities used for instantiation of each red agent within each of the types of red agent

units discussed above. Figure 21 depicts the initial starting locations for the entire environmental template of the enemy force used in these experiments.

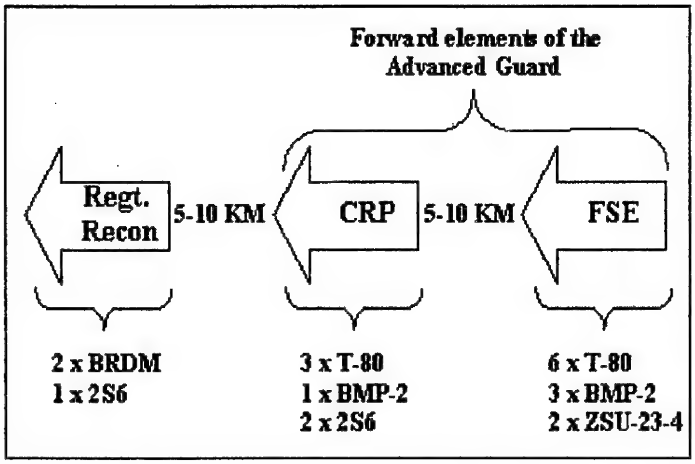


Figure 20. Forward recon elements of a Soviet MRR.

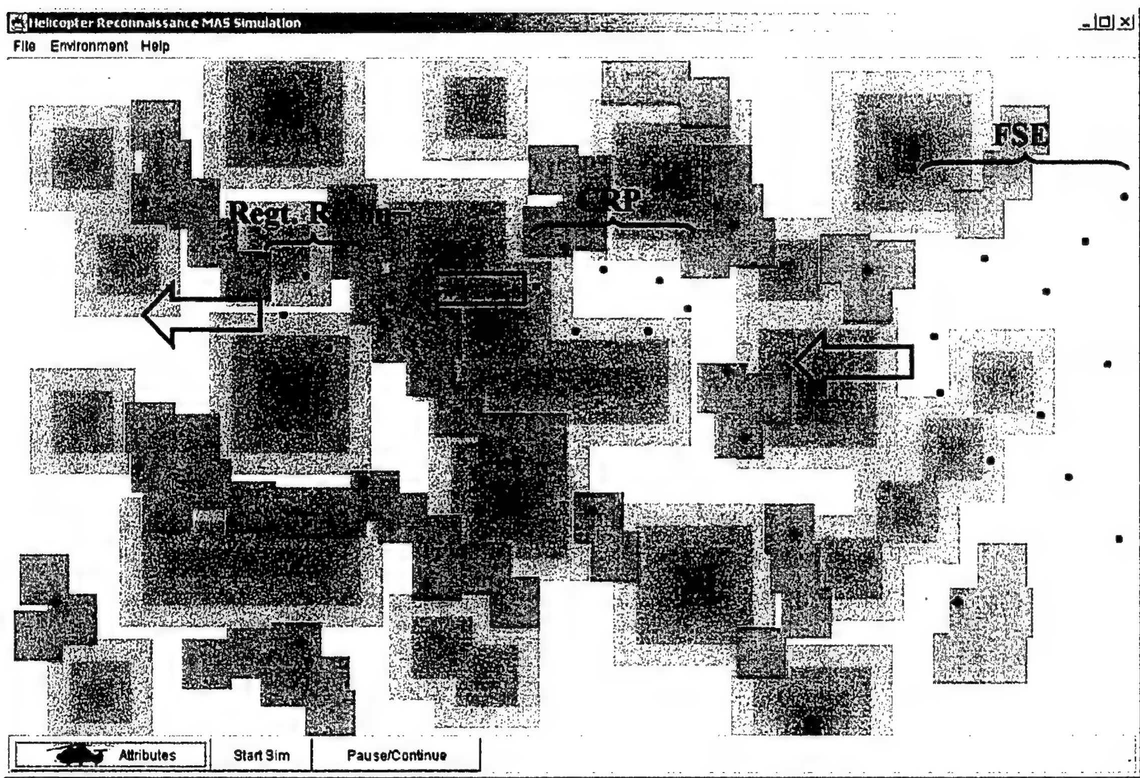


Figure 21. Environment used in experimental analyses.

System Attributes					
Type	Quantity	Speed KPH	TOS (min)	Wpn Credits	Sensor Rng (m)
BRDM	2	20	240	10	8,000
2S6	1	20	240	10	10,000
T-80	3	15	180	10	6,000
BMP-2	1	15	210	10	6,000
2S6	2	15	240	10	10,000
T-80	6	10	180	10	6,000
BMP-2	3	10	210	10	6,000
ZSU-23-4	2	10	210	10	10,000

Table 2. Experiment One: red agent system attributes.

Movement Propensities				
Unit	Shortest Distance	Use of Terrain	Use of Cover	Avoid Enemy
Regt. Recon	4	3	3	5
CRP	4	5	3	4
FSE	4	5	3	2

Table 3. Experiment One: red agent movement propensities.

5. Experiment One

The first experiment incorporates five blue agent profiles to represent the full spectrum of U.S. Army Aviation tactical flight profiles used when conducting aerial reconnaissance. The blue team attributes, aircraft attributes, and tactical movement propensities are subjectively set to realistically represent plausible characteristics

respective with its profile. See Tables 4 and 5 for depiction of the attributes and propensities used for instantiation of each of these blue agent team profiles. The flight profile, speed, thoroughness of reconnaissance (reconnaissance LOD), and sensor range are the primary attributes that doctrinally vary within employment of these tactical flight profiles. Therefore, these are the only attributes that differ between the tactical flight profiles implemented for this experiment. Note that the tactical movement propensities are kept the same for all profiles.

Team & Helicopter Attributes							
Expt Team Name	**Profile**	**Recon Spd (kph)**	**Recon LOD**	**TOS (min)**	**Missiles**	**Rnds**	**Sensor Rng (m)**
NOE #1	NOE	45	7	120	8	500	4,000
NOE #2	NOE	70	5	120	8	500	5,000
CTR #1	Contour	95	4	120	8	500	6,000
CTR #2	Contour	120	3	120	8	500	7,000
LL	Low Level	145	1	120	8	500	8,000

Table 4. Experiment One: blue agent team and helicopter attributes.

Movement Propensities				
Expt Team Name	**Shortest Distance**	**Use of Terrain**	**Use of Cover**	**Avoid Enemy**
All 5 Profiles	2	5	4	3

Table 5. Experiment One and Two: blue agent movement propensities.

Each of these five profiles is then run through the scenario's environment and red agent enemy forces for 50 replications. The random repositioning of initial red agent starting locations is selected to implement more red agent movement variance throughout the terrain.

All eight summary statistics are captured via *out* files for each of the 50 individual replications for each of the five types of profiles. These five profiles are then analyzed to determine which profiles performed most successfully with regard to five of the summary statistics. The five summary statistics used to define success are: the fewest undetected red agents, most red agents hit, most red agents killed, and fewest blue agents hit and killed in proportion to the number of blue agent helicopters used to reconnoiter the sector.

It is important to reiterate that the number of undetected red agents is defined as the number of red agents successfully reaching their main objective, plus the number of fully operational red agents still within the environment when the blue agents have completed reconnaissance of the sector. The number of red agents hit is defined as the number of red agents that are initially hit but not killed. The number of red agents killed may include red agents that were previously hit, but subsequently killed. Additionally, the numbers of blue agents hit and killed are single helicopters. Recall that a blue agent team consists of two helicopters (tandem).

6. Experiment Two

The second experiment incorporates three blue agent profiles that implement the blue agent attributes determined to be the most influential characteristics responsible for blue agent success in Experiment One. The area of interest in Experiment Two is to determine if minimal and realistic adjustments of the four differing blue agent attributes

from Experiment One could produce a tactical profile that captures the overall success experienced by all five profiles from Experiment One. Experiment Two demonstrates the model's ability to capture and reflect the impact of blue agent attributes and characteristics with respect to the terrain and red agent template created for these experiments. This experiment uses three separate modified profiles from Experiment One that are run for 50 replications with summary statistics collected via *out* files. All other experimental parameters are identical to those of Experiment One. See Tables 5 and 6 for depiction of the attributes and propensities used for instantiation of each of these blue agent team profiles.

Team & Helicopter Attributes							
Expt Team Name	Profile	Recon Spd (kph)	Recon LOD	TOS (min)	Missiles	Rnds	Sensor Rng (m)
NOE #3	NOE	75	6	120	8	500	5,000
NOE #4	NOE	90	5	120	8	500	6,000
NOE #5	NOE	110	4	120	8	500	7,000

Table 6. Experiment Two: blue agent team and helicopter attributes.

7. Logistical Synopsis

The final analysis incorporates the 50 replications of all eight profiles from Experiments One and Two to show the ammunition and asset logistical requirements utilized during the experiments. The area of interest in this logistical synopsis is to demonstrate the model's initial ability to capture and depict logistical requirements during model replications.

C. RESULTS AND ANALYSIS

This paragraph discusses the results and method of analyses of the two experiments and logistical synopsis previously discussed. Experimental model results are presented using line graphs depicting means and standard errors for each of the similar cases considered. Appendix B contains additional descriptive statistics of the data obtained from these tactical profiles and model replications. Figure 22 depicts the statistical formulae used to obtain the means and standard errors.

$$M = \frac{\sum_{i=1}^{n_y} y_{ip}}{n_y} \qquad S.E. = \sqrt{\frac{\sum_{i=1}^{n_y} (y_{ip} - M)^2}{(n_y - 1)(n_y)}}$$

The parameters have the following meaning:

n_y = number of replications for each profile
i = replication number
p = tactical profile
y_{ip} = data value
M = arithmetic mean
S.E. = standard error

Figure 22. Mean and standard error formulae.

1. Experiment One

The statistics produced by Experiment One resulted in very plausible values and are consistent with current U.S. Army Aviation thinking concerning tactics, techniques, and procedures. U.S. Army Aviation tactical flight training emphasizes the use of terrain and cover to enhance survivability and hinder the enemy's detection capability. These skills are trained via NOE flight profiles that implement lower altitudes and slower

airspeeds while using terrain and vegetation as cover and concealment. The results of Experiment One enforce this instructed and trained technique.

The major findings of Experiment One show that NOE profiles resulted in significantly better blue agent survivability, Contour profiles resulted in significantly more mission success, and the Low Level profile was inferior in both respects.

Figures 23, 24, and 25 show the overall mean number of undetected red agents, red agents hit, and red agents killed during the 50 replications for each of the five tactical profiles. These figures clearly show the significantly better performance associated with the Contour flight profiles and much poorer performance associated with the Low Level profile.

Note that Contour #1's interval of mean plus and minus one standard error does not overlap with any of the NOE profiles. This suggests that the differing attributes (speed, reconnaissance thoroughness, and sensor range) have an impact on mission success. Additionally, note that the Low Level profile interval of mean plus and minus one standard error does not overlap with any other profile standard errors. This clearly indicates the poor mission success associated with the Low Level profile.

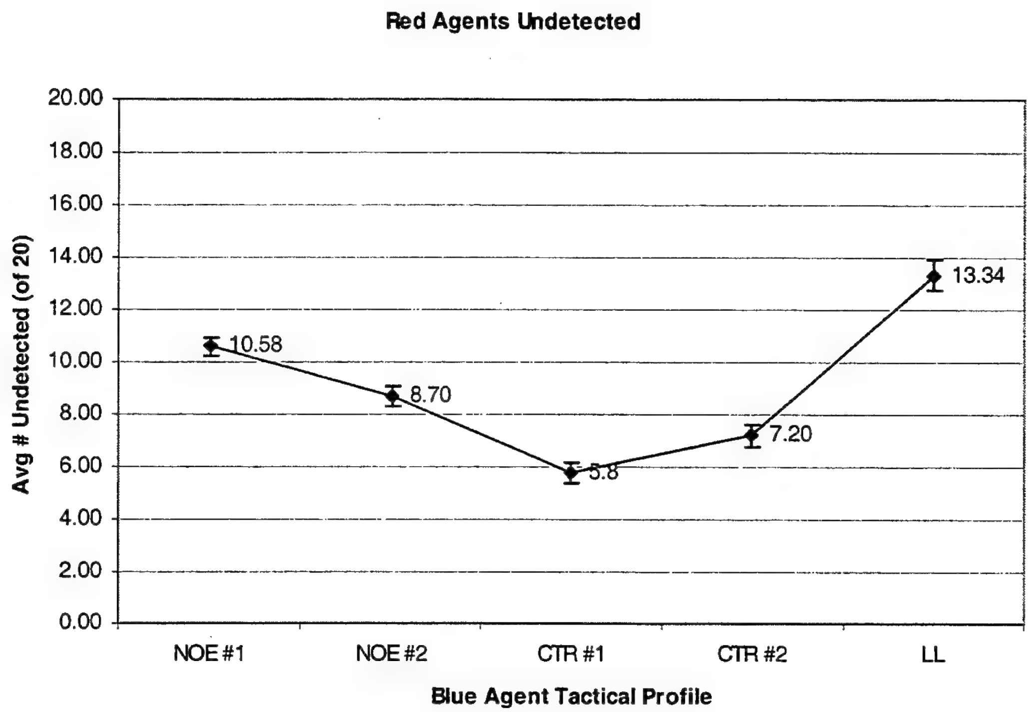


Figure 23. Red agents undetected in Experiment One.

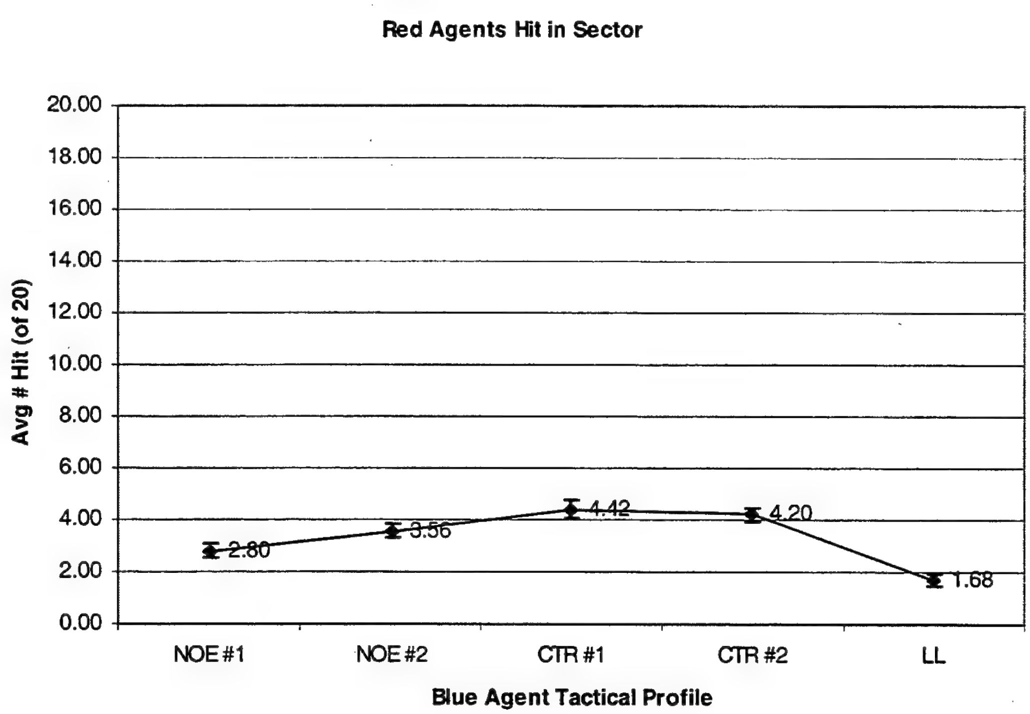


Figure 24. Red agents hit in sector in Experiment One.

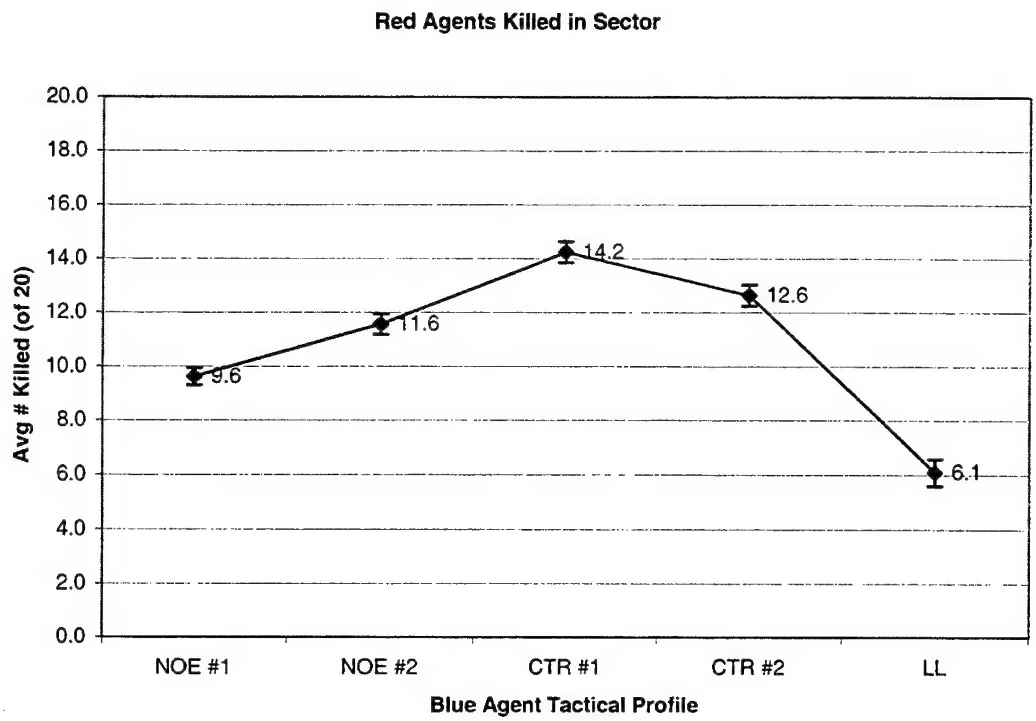


Figure 25. Red agents killed in sector in Experiment One.

Figures 26 and 27 show the mean percent of blue agents hit and killed during the 50 replications for each of the five tactical profiles. These graphs specifically depict the mean percent of agents hit and killed of the total number of aircraft used for each profile. For the ordering of the profiles displayed, the lines connecting the profile means show an increasing relationship between the profiles and their survivability. Although consecutive profile intervals of mean plus and minus one standard error overlap, there is significantly better overall performance associated with NOE profiles compared to Contour #2 and Low Level profiles, and the Contour #1 profile compared to the Low Level profile. Note once again the significantly poorer survivability performance associated with the Low Level profile. This is consistent with the tactical flight training

techniques previously discussed. The increased speed and sensor range associated with the Low Level profile does not result in improved mission success and survivability. The use of terrain and vegetation as cover and concealment play an integral role in the performance of helicopter teams conducting armed reconnaissance.

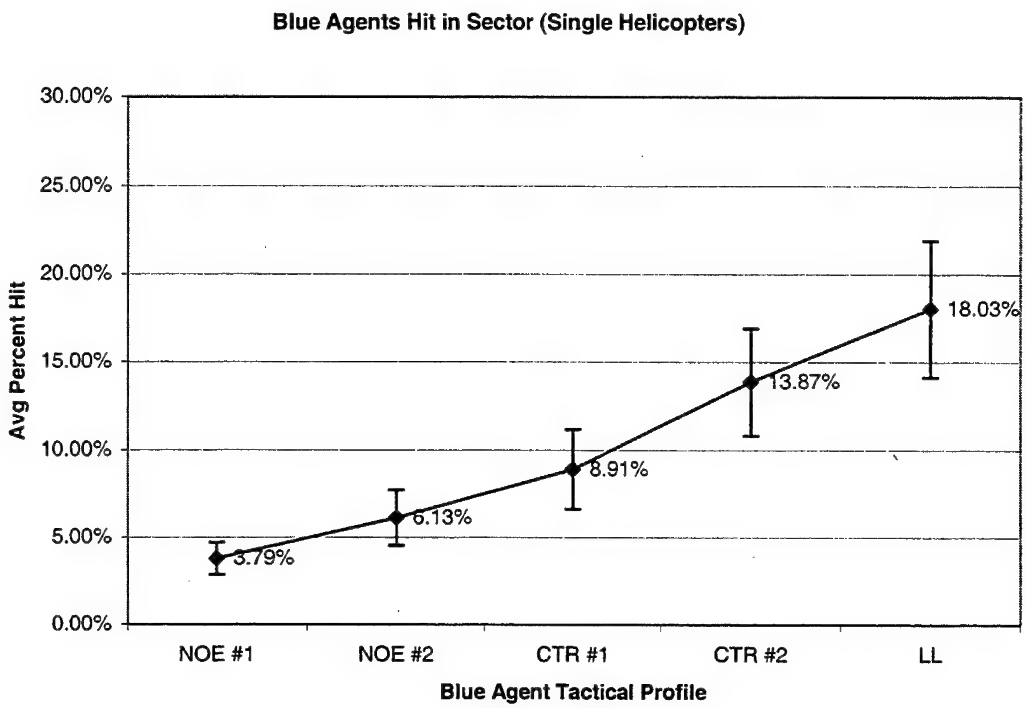


Figure 26. Blue agents hit in sector in Experiment One.

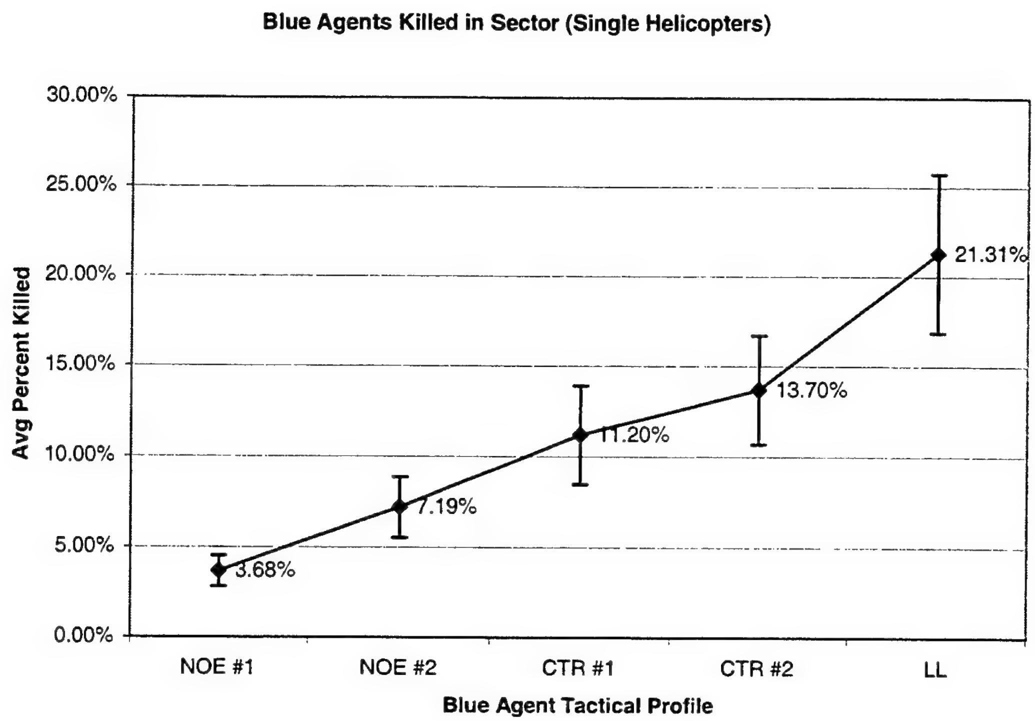


Figure 27. Blue agents killed in sector in Experiment One.

2. Experiment One Discussion

In summary, the results of Experiment One indicate that NOE profiles provide better blue agent survivability, while Contour profiles result in a smaller mean number of red agents that are undetected and a greater mean number hit and killed. Thus, a Contour profile results in more mission success. Given that only the flight profile, speed, reconnaissance thoroughness, and sensor range differed amongst the profiles, the subsequent objective of Experiment Two was to determine which of those attributes accounted for the success within each type of profile. Since the Low Level profile performed poorly in all aspects, it is not discussed in any further experimental analyses.

It was assumed that the survivability difference between NOE and Contour is accounted for by the actual profile attribute. Therefore, NOE is implemented as the only

profile for analysis in Experiment Two. The significance of the other three attributes on mission success was less obvious. Since NOE is chosen as the profile for Experiment Two, the modifications made to speed, reconnaissance thoroughness, and sensor range have to be realistic and within the tactical performance capabilities of the NOE profile. Therefore, only minimal increments are added to these attributes in attempt to produce better mission success and maintain the survivability experienced with the NOE profile. Table 6 depicts the three tactical profiles created from Experiment One for further analysis in Experiment Two. Note again that all other experimental parameters, including movement propensities, remain the same. The hypothesis is that these newly created NOE profiles will result in continued survivability success, while minimal increments to speed, reconnaissance thoroughness, and sensor range will improve mission success, but not necessarily linearly.

3. Experiment Two

The statistics produced by Experiment Two show the hypothesis to be true. Once again, the experiment produced plausible values and the results are consistent with current U.S. Army Aviation thinking concerning tactics, techniques, and procedures.

The major findings of Experiment Two show that the NOE profile experienced nearly the same blue agent survivability success experienced in Experiment One, and with minimal increments to the other three attributes, mission success was improved to values near those of the Contour profiles of Experiment One. It is interesting to note that continued enhancements to the other three attributes did not necessarily translate into linearly related mission success. This suggests that blue agent team tactical profiles also play an integral role in mission success. The limiting factors associated with using

terrain and vegetation during reconnaissance cannot necessarily be overcome by increasing the speed and sensor range.

Figures 28, 29, and 30 show the overall mean number of undetected red agents, red agents hit, and red agents killed during the 50 replications for each of the three NOE profiles, along with the original five profiles used in Experiment One. These figures show mean numbers of undetected, hit, and killed red agents that are much closer to the results produced by the Contour profiles from Experiment One.

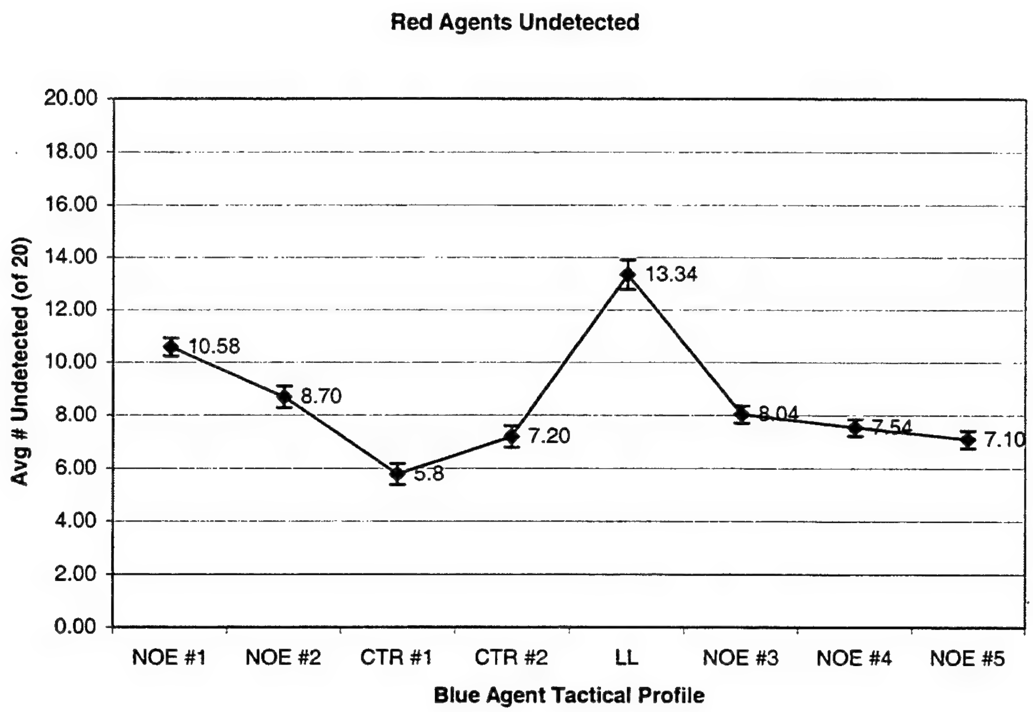


Figure 28. Red agents undetected in sector in Experiment Two.

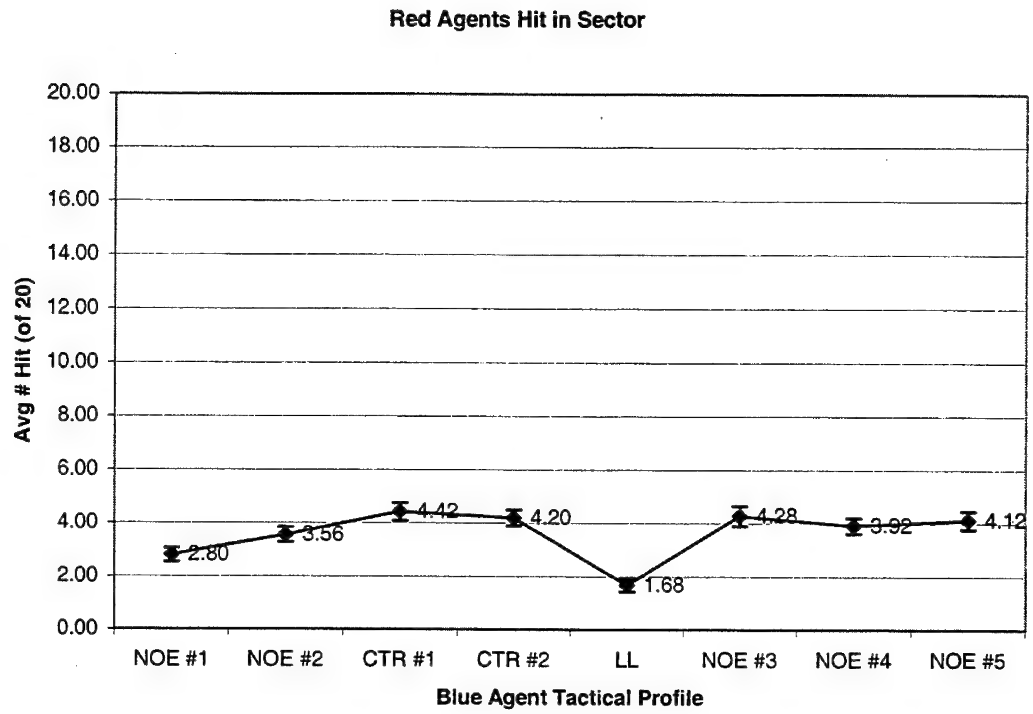


Figure 29. Red agents hit in sector in Experiment Two.

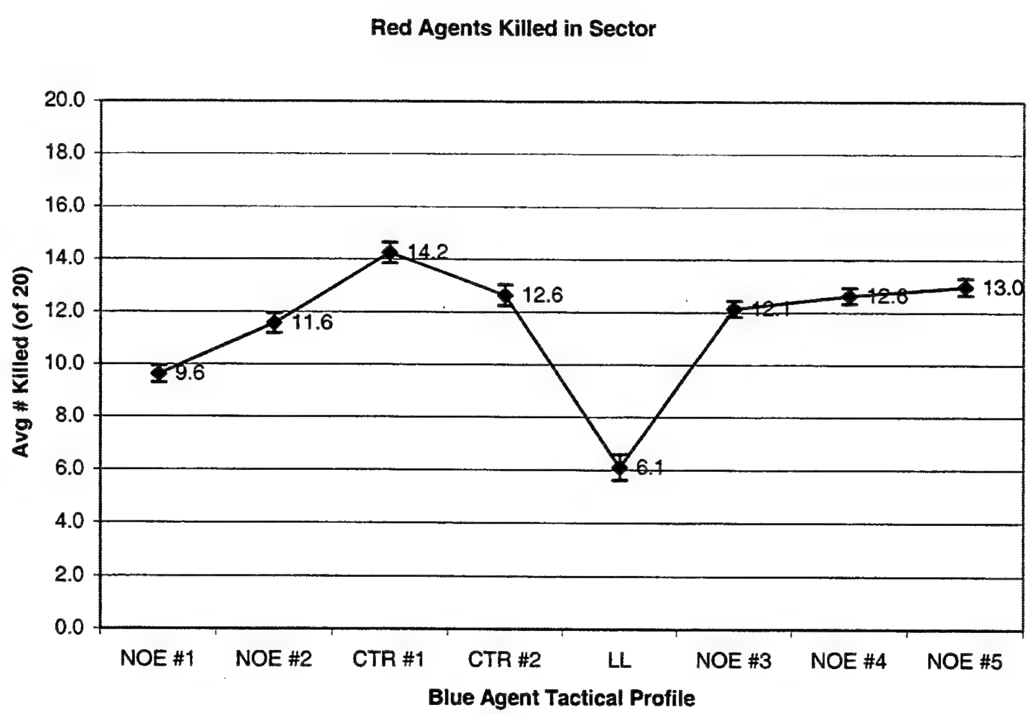


Figure 30. Red agents killed in sector in Experiment Two.

Figures 31 and 32 show the mean percent of blue agents hit and killed during the 50 replications for each of the three NOE profiles, along with the original five profiles used in Experiment One. These figures show mean percentages of blue agents hit and killed for the NOE #3 and NOE #4 profiles nearly achieve the survivability success achieved with the NOE profiles from Experiment One.

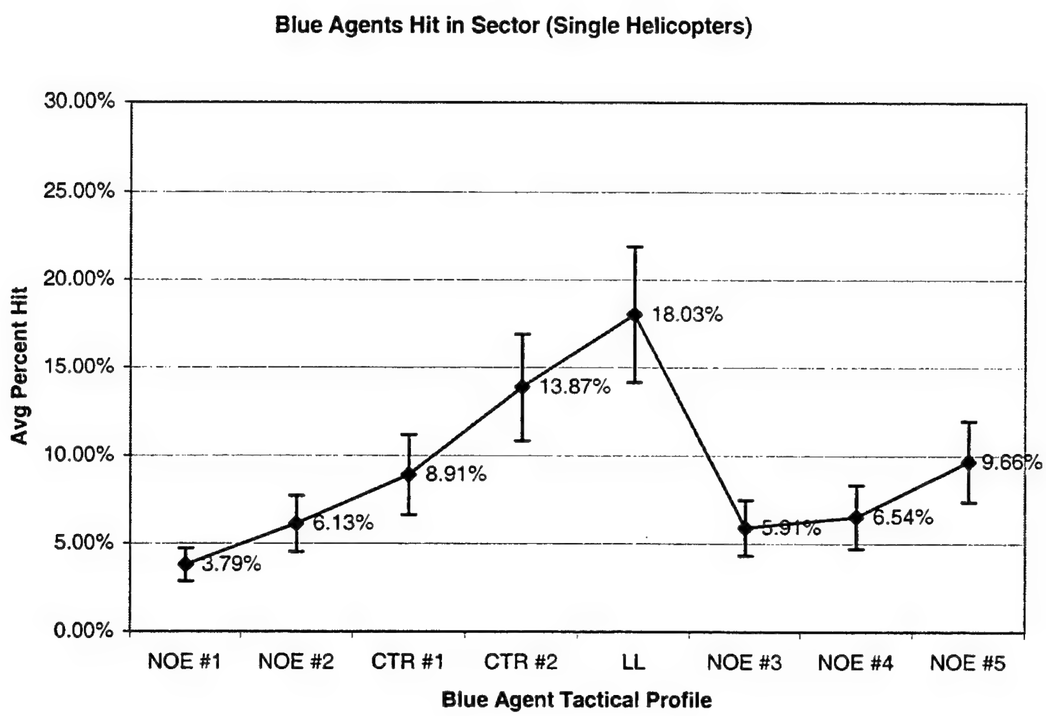


Figure 31. Blue agents hit in sector in Experiment Two.

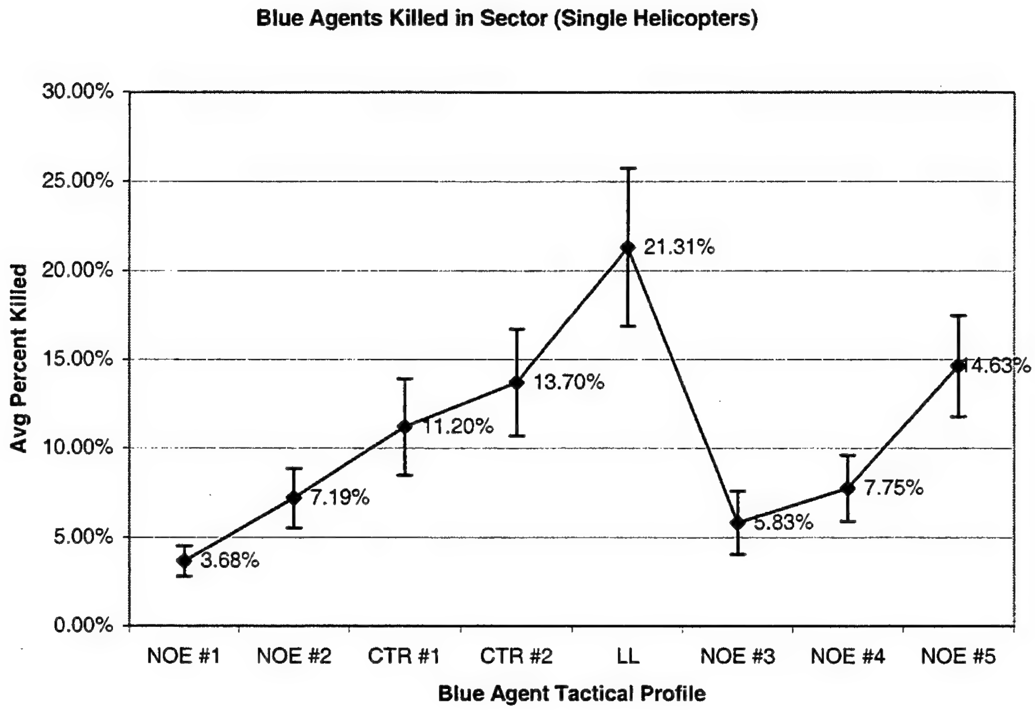


Figure 32. Blue agents killed in sector in Experiment Two.

4. Experiment Two Discussion

In summary, the results of Experiment Two support the initial hypothesis that NOE profiles result in continued survivability success. Additionally, minimal increments to speed, reconnaissance thoroughness, and sensor range improve mission success, but not in a linear manner.

Figures 28, 29, and 30 show how small increments in speed, reconnaissance thoroughness, and sensor range enabled the blue agent teams to more successfully find and kill the enemy red agents overall. More importantly, continued increments of these attributes do not result in a significant improvement amongst the three success criteria. NOE #5 does have the largest mean number of red agents killed when compared with NOE #3 and NOE #4, but it also has the largest mean percentage of helicopters killed.

Additionally note the significant mean plus and minus one standard error overlap for all three profiles depicted in Figures 28 and 30. There is no clear improvement amongst the profiles as the attributes are enhanced. There appears to be a tradeoff between mission success and survivability.

The underlying factor for Contour #1's better mission success is a result of the model's limitation of only one blue team conducting reconnaissance at a time. The Contour profile's advantage in speed enabled teams with this profile to cover the sector faster, thus resulting in the blue agent's ability to find red agents sooner and develop the situation deeper within the sector before red agents were able to reach their main objectives. This insight was made during observation of the different profile replications. This model limitation is discussed in more detail in Chapter 5. Although this limitation is a factor, it does not eliminate the evidence that these attributes do not linearly increase the blue agent performance during reconnaissance.

Figures 31 and 32 again support the evidence that the tactical profile implemented by blue agent teams is critical. The NOE #3 and NOE #4 profiles clearly indicate that survivability success is maintained by using the NOE tactical profile. Although there appears to be a slight decrease in mean survivability with these new profiles, the decrease is not significant due to the overlap of standard errors. The underlying factor for the slight decrease in survivability is explained by the increased interaction between the blue and red agents. The improved blue agent detection and destruction of red agents results in more agent interaction, hence increased chances for red agents to acquire and engage blue agents.

The most interesting outcome of Experiment Two is the realistic tradeoff presented to the user when regarding mission accomplishment and survivability. This is true in all aspects of combat. The underlying question that will always exist: "What combination of survivability and aggressiveness will ultimately provide the commander with the best acceptable outcome?" Experiment Two demonstrates that this model can realistically be used to analyze and modify profiles and attributes inherent to helicopter armed reconnaissance. Produced results exhibit consistency with the tactics, techniques, and procedures employed by U. S. Army Aviation doctrine.

5. Logistical Synopsis

The logistical statistics gathered by the model include: the mean number of blue agent teams used to conduct reconnaissance of the sector, and mean expenditures of missiles and gun rounds. Obviously these quantities are direct reflections of the type of profile employed, speed of reconnaissance, endurance, and enemy interaction. The intent of these graphs is to display the logistical impact these various profiles create and the ability of the model to portray those quantities. These quantities can provide planners and leaders with insight of logistical and asset requirements and trends with respect to the attributes and scenario being analyzed.

Figures 33, 34, and 35 show the overall mean number of blue agent teams used to reconnoiter the sector, number of missiles expended, and number of rounds expended for the 50 replications for each of the eight profiles.

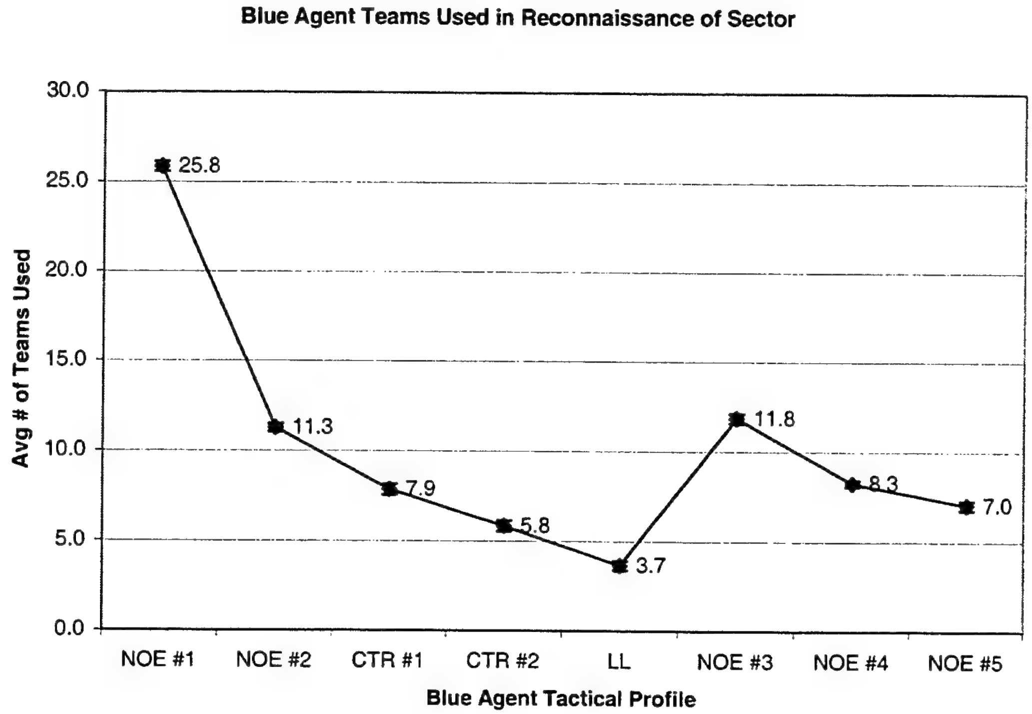


Figure 33. Blue agent teams used to reconnoiter sector.

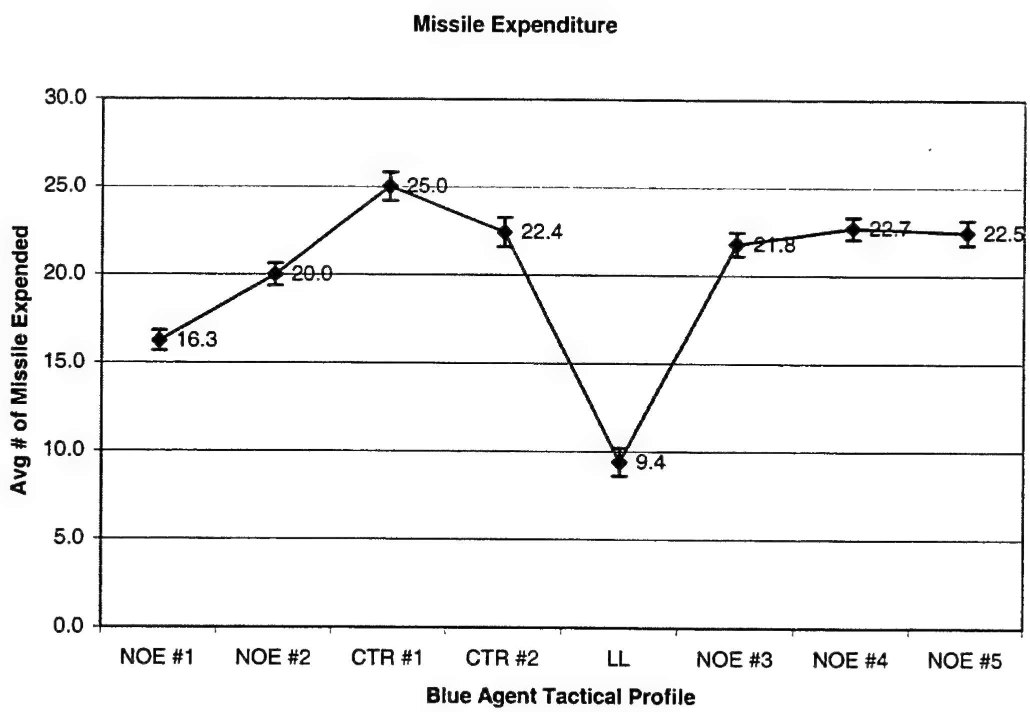


Figure 34. Missile expenditure during sector reconnaissance.

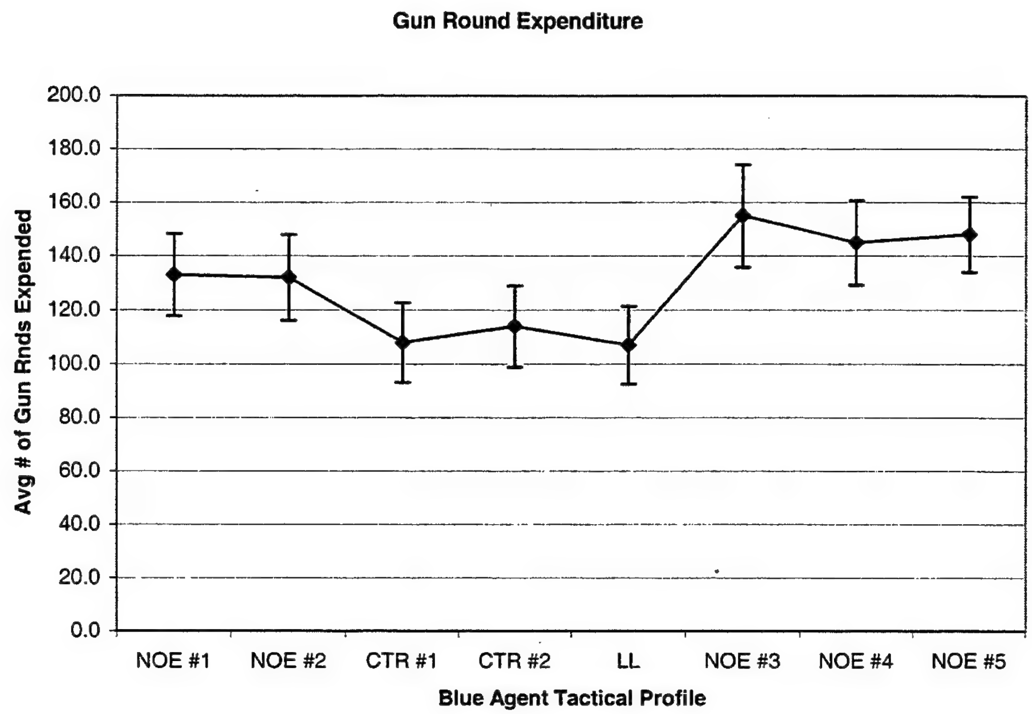


Figure 35. Gun rounds expenditure during sector reconnaissance.

6. Logistical Synopsis Discussion

In summary, the results of the logistical synopsis demonstrate the model's ability to capture and depict primary logistical considerations. Although these quantities are primarily direct reflections of the instantiated helicopter and team attributes, there are some noteworthy insights. Blue agent team and helicopter attributes do not only affect mission success and survivability. They additionally impact the number of teams and ammunition required.

The total number of blue teams used represents the number of team ROSs required to cover the sector with respect to endurance limitations and hit/killed teams. The number of blue teams required when using the NOE #1 profile is much greater than the other NOE profiles. Recall that all profiles have the same endurance and movement

propensities. The NOE #1 speed is only 25 KPH (13 KTPH) less that of NOE #2, yet it required more than twice the number of teams to complete the reconnaissance, and experienced significantly less mission success. This suggests that the combinations of blue agent attributes produce additional ramifications for the management of blue team assets. The minimal gain in survivability achieved with the NOE #1 profile is offset by a decrease in mission success and increase in team asset requirements. There appears to be a tradeoff associated with a minimal gain in survivability versus substantially more blue agent team rotations and less success with respect to finding and killing the enemy.

Another interesting depiction of Figure 33 is its relationship with the number of blue helicopters hit and killed (Figures 31 and 32). This especially reiterates the poor performance associated with the Low Level profile teams. The graphs together more clearly depict how disproportional the blue agent hits and kills are distributed amongst the different profiles. The use of terrain, cover, and concealment clearly plays an integral role in the results produced by this model.

Observation of the descriptive statistics in Appendix B shows that all profiles averaged between one and two kills per model replication. This is interesting when comparing with mission success and percentages of blue agents killed. Although the asset losses are nearly the same, a critical tradeoff is highlighted when comparing with the successful accomplishment of the mission and the percentage of total assets hit/killed.

As previously stated, Figures 34's and 35's ammunition expenditures are a direct reflection of the amount of agent interactions and engagements. Obviously blue agent ammunition expenditures increase as their success improves in finding the enemy. The insightfulness gained from these graphs is the summarized overview of ammunition

requirements. Even without regard to the profile used, this scenario's summarized results indicate that the majority of the model replications could be successfully accomplished with a mean of 22 missiles and a mean of 140 rounds of ammunition (plus and minus one standard error).

The reader is cautioned that the P(H), P(K), and weapon ranges were obtained from the U.S. Army's JANUS simulation [JANUS, 1999] (Appendix A). These ammunition quantities should not be used for actual prediction purposes. The program's computer code should be modified accordingly to properly reflect applicable helicopter weapons systems performance parameters and intents of usage. Additionally, only one team is in sector conducting reconnaissance at a time. A future enhancement that enables a user to divide the sector and have multiple teams conducting reconnaissance will provide additional insight into the operational performance and logistical requirements. This future enhancement is discussed in more detail in Chapter 5.

D. FINAL DISCUSSION

The experiments presented in this chapter were intended to demonstrate the model's ability to produce realistic and plausible results that are consistent with U.S. Army Aviation tactics, techniques and procedures. The model certainly possesses limitations that hinder its ability to represent all tactical aspects and employment techniques. But it should be kept in mind that this model is an initial simulation tool. Future enhancements and modifications will only add to the usefulness and capabilities of this model. Suggestions for future work, enhancements, and modifications are subsequently addressed in Chapter 5.

All blue agent teams used in these experiments implemented the Comanche platform type. Following official verification and validation of this model, future model enhancements and modifications may enable this model to be used in comparison analyses of different platform types. It was felt that the model characteristics used to represent the Comanche helicopter did not fully represent its proposed design capabilities. Premature comparison at this point in the model's development would not truly depict the full spectrum of the Comanche helicopter's systems.

As previously discussed, only one terrain and red agent enemy scenario is used throughout these experiments. It is impossible to develop and analyze all of the environments that might be created for analyses. The model was created with great versatility, but only explored for one. It is left to the user to create the terrain, enemy red agents, and blue agent characteristics of interest for his or her analyses.

In summary, this chapter demonstrates the model's potential. With the implementation of the enhancements and capabilities discussed in the next chapter, this model can ultimately result in a simulation tool that is integral to the T&E phase of the Comanche helicopter development cycle.

V. FUTURE WORK AND CONCLUSION

A. FUTURE WORK

This section describes some possible future enhancements and modifications to this model. Many of these enhancements would add to the realistic representation of helicopter armed reconnaissance and provide Comanche T&E combat developers with better simulation features necessary for logistical forecasting, operational evaluation, and helicopter-platform comparison analyses.

1. Division of Sector with Additional Blue Agent Teams

One model limitation noted in Chapter 4 was the current inability to have more than one blue agent team conducting reconnaissance in sector. To overcome this limitation, the GUI could be modified to allow a user to divide the environment and assign reconnaissance subsections (zones) to additional blue agent teams. This enhancement would add an interesting area of analysis with regard to how blue agents perform when delegating the area amongst additional teams. Ultimately it would provide the user the ability to control the number of blue agent teams in sector with the aim of producing superior mission success and unit survivability.

2. Implementation of More Advanced Systems Attributes

The ultimate objective of this model is the development of a simulation tool that assists Comanche T&E personnel (and others with an analogous mission) with logistical requirements forecasting, and to conduct comparisons with the Kiowa Warrior. The current state of the model does not capture all of the key system capabilities proposed for implementation on the Comanche Platform. Many of the advanced systems being

developed for the Comanche are still in the phase of engineering development during the period of research for this thesis. As these systems mature, their parameters can be integrated into the model to support further analyses in specific areas of interest, performance, and to compare with previous or current platforms.

Key advanced Comanche systems critical to the enhancement of this model include: the fire control radar, maintenance failure rates for the systems and airframe, advanced Hellfire missiles, specific performance parameters of the 20mm gun, digital communications capability, second generation Forward Looking Infrared (FLIR), and its appropriate detection signature. With the implementation of these attributes, analysts will be better able to measure the systems effects of the Comanche, and to compare its operational performance with that of the Kiowa Warrior.

3. Failures, Maintenance, and Their Impact on Operational Performance

Maintenance plays an integral part in helicopter readiness and availability. The inclusion of sensor, weapon, and aircraft maintenance failure rates would add other parameters for further realistic analyses and performance comparison. Maintenance failure rates would enable analysts and leaders to evaluate required blue agent assets, address maintenance logistical requirements, and identify operational limitations.

Maintenance failure rates could be easily incorporated into this model through the inclusion of proper statistical distributions and stochastic processes. Incorporating settable parameters for failure and repair times of various systems require implementation that enables more complete analysis of the operational performance and impact. For more in-depth discussion on this topic, see Kevin Schmidt's master's thesis that is concerned with simulating operational sensitivity of the Comanche [Schmidt, 1999].

4. Agent Relationships and Communication

Primary elements of MASs are the relationships agents form, combined with the communication structure within those relationships. The current state of the present model only implements one level of relationship for each type of agent. Blue agents represent a total reconnaissance team entity, and red agents operate individually, with no leadership or unit integrity. Implementation of a hierarchical unit structure with imbedded relationships could better represent the influence of unit cohesion, orders, and leadership intent.

Future agent relationship structures and units might include companies, battalions, the Tactical Operations Center (TOC), and commanders. These types of relationships would add many different dimensions to the model for analysis. These enhancements would enable a user to input the cognitive aspects involved with leadership and management and then observe their influence on operational performance.

Red agent unit relationships and inter-agent communication could enable agents to devise strategies, implement deception, and communicate opponent agent interactions with other same-type agents. Ultimately such extensions would result in more intelligent agents that could strategically structure their movement and adjust main objectives while moving through the sector.

5. Addition of More Objects

This model currently only possesses a few of the many terrain features depicted on navigational maps. The inclusion of additional terrain features would enable a user to create environments of greater detail. These features would further impact the movement propensities and paths selected by agents during tactical movement. Some additional

terrain features might include: depressions, man-made structures, urban areas, roads, rivers, lakes, and man-made obstacles. These features could be implemented in the model via the same techniques used for the current terrain features.

An additional feature that was implemented for red agents, but not for blue agents is re-supply cache sites. The inherent capabilities associated with ground vehicle red agents require the integration of re-supply cache sites for their continued movement through sector. This capability was not necessary for blue agents because they must return across the LD to a non-depicted assembly area following ROS. The ability to instantiate Forward Arming and Refuel Points (FARPs) within the sector might be integrated to enable blue agent re-supply during reconnaissance. A further enhancement might allow these FARPs to become active at dynamic times of reconnaissance progress. This feature would add obvious impacts on mission performance, the required number of blue agents used during a scenario, and platform TOS. The vulnerability of either side's re-supply cache sites can be a factor in determining either's mission success, and resource requirements.

It should be noted that this current FARP limitation does create a critical logistical impact on performance under certain conditions, especially for the Kiowa Warrior. That impact was observed during model runs using Kiowa Warriors with minimal endurance. Realistic endurance limitations of the Kiowa Warrior (especially with weapon loads) often hinder its ability to conduct deep missions without forward positioned FARPs. These limitations are observed when instantiating blue teams with minimal TOS. The result is blue agent teams that spend all of their TOS returning to and from their ROS

checkpoint. This limitation does denote a critical observation in the comparison analyses between the Kiowa Warrior and the Comanche's extended endurance capability.

6. Use of Genetic Algorithms

The use of genetic algorithms could enable analysts to specifically analyze the effects of key parameters of interest. Genetic algorithms provide a method for determining optimal parameter settings to obtain a certain level of performance. As in biology, and specifically genetics, specific helicopter characteristics and team attributes could be represented with alleles. In biology, alleles contain the specific descriptive characteristics that define a chromosome. These allele representations could be used to encompass the full range of a specific team or helicopter attribute. The total make-up of a team's attributes would represent the entire team's chromosome. By randomly assigning these alleles to blue agent teams the full spectrum of a single attribute's performance could be observed. An evaluative-type system could then be applied that rewards top-performing alleles and degrades poorly performing alleles. Ultimately, top-performing alleles approach a steady state and identify themselves as the best performing attribute setting for that given parameter and scenario.

Genetic algorithms present a very powerful tool for exploring agent-based modeling. For a better understanding of this concept, the reader is encouraged to read John Holland's book titled "Hidden Order" [Holland, 1995]. This book provides an in-depth method for integrating genetics with agent-based modeling to explore the power of agent adaptation and system complexity.

7. Enhanced Engagement and Detection Algorithms

As previously pointed out in Chapter 3, the engagement and detection algorithms in this model serve as placeholders. More advanced algorithms for detection, probability of hit (P(H)), and probability of kill (P(K)), may be incorporated. Enhancements to the detection algorithm might include probabilities of recognition, correct (or incorrect) classification, and identification. This enhancement could potentially introduce the consequences of fratricide and target misidentification, ultimately affecting accuracy of information gathering, mission success, and performance. Further enhancements to shooting might include more detailed P(H) and P(K) data, and weapon limitations associated with ammunition failures, human error, and aiming errors.

8. Environmental Considerations

The current model is constructed within a two-dimensional environment that incorporates an artificial third dimension to represent elevation. This enables a user to create terrain and agents that are instantiated with respect to their three-dimensional characteristics. A very comprehensive modification to the model could be developed to allow the importation of three-dimensional terrain files. This modification would allow users to develop scenarios using actual terrain maps represented visually through computer graphics editing tools. This method of three-dimensional terrain integration is demonstrated in Jason Stine's master's thesis research that is concerned with expert land navigation [Stine, 2000].

An additional environmental consideration is the implementation of night operations. A simplistic approach to this enhancement should involve adjustable parameters for selecting day/night and setting agent sensor/weapon ranges applicable to

night operations. This attribute could enable the analysis of agent performance with respect to advanced target acquisition systems and night vision imagery such as FLIR.

9. Tactics, Techniques, and Procedural Considerations

The initial model has been developed to explore armed reconnaissance. Agent engagements are automatically adjudicated given a successful detection. A key aspect of reconnaissance is finding and gathering information concerning the enemy. This requirement does not necessarily require interaction and engagement with the enemy. An information-gathering enhancement could be implemented to explore the blue agent ability to merely find and observe enemy operations. This could be further developed with the communication enhancements discussed above to explore the leadership vision of the enemy situation and intent.

Attack operations could also be implemented. Additional attack assets are usually moved forward to destroy and disrupt enemy movement once significant enemy strongholds are detected. These additional attack assets could be integrated through the direction of blue agent helicopter teams or ground maneuver agents representing blue armor and mechanized vehicles.

B. CONCLUSION

This thesis articulates the modeling of helicopter armed reconnaissance through agent-based modeling. The model developed for this thesis demonstrates how agent-based modeling can capture many of the cognitive and tactical aspects of helicopter armed reconnaissance. Additionally, the current model produces results consistent with U.S. Army Aviation tactics and offers many beneficial analytical opportunities.

As previously discussed, there are many areas for potential future work on, and enhancement to, this model with respect to tactics, advanced systems attributes, and environmental considerations. Continued integration of these enhancements will only add to the usefulness and capabilities of this simulation tool.

With continued work in this area of research and model development, this model will ultimately provide the Comanche helicopter T&E combat developer with the simulation tool required for logistical forecasting, operational evaluation, and helicopter-platform comparison analyses.

APPENDIX A. JANUS VERSION 7.06D PROBABILITY OF HIT AND KILL PERCENTAGES

500m min range	8000m max range	Target Conditions
0.55	0.45	Stationary & Defilade
0.95	0.85	Stationary & Exposed
0.85	0.80	Moving & Exposed

Table 7. Hellfire probability of hit/kill percentages. From Ref. [JANUS, 1999].

Shooter Conditions	0m min range	1500m max range	Target Conditions
Stationary	0.0525	0.035	Stationary & Defilade
Stationary	0.105	0.07	Stationary & Exposed
Stationary	0.084	0.056	Moving & Exposed
Moving	0.042	0.028	Stationary & Defilade
Moving	0.084	0.056	Stationary & Exposed
Moving	0.063	0.003	Moving & Exposed

Table 8. 30mm probability of hit/kill percentages. From Ref. [JANUS, 1999].

0m min range	3750m max range	Target Conditions
0.75	0.75	Stationary, Exposed, & Flank
0.65	0.65	Stationary, Exposed, & Head-On

Table 9. T-80 tank AT-11 probability of hit/kill percentages. From Ref. [JANUS, 1999].

2400m min range	8000m max range	Target Conditions
0.20	0.20	Stationary, Exposed, & Flank
0.15	0.15	Stationary, Exposed, & Head-On
0.40	0.40	Moving, Exposed, & Flank
0.30	0.30	Moving, Exposed, & Head-On

Table 10. 2S6 ADA SA-19 probability of hit/kill percentages. From Ref. [JANUS, 1999].

0m min range	3000m max range	Target Conditions
0.05	0.025	Stationary & Defilade
0.09	0.05	All Others

Table 11. ZSU-23 ADA gun probability of hit/kill percentages. From Ref. [JANUS, 1999].

65m min range	4000m max range	Target Conditions
0.675	0.675	Stationary, Exposed, & Flank
0.675	0.45	Stationary, Exposed, & Head-On

Table 12. BMP-2 AT-5 probability of hit/kill percentages. From Ref. [JANUS, 1999].

0m min range	1400m max range	Target Conditions
1.00	1.00	All Conditions

Table 13. BRDM-2 14.5mm probability of hit/kill percentages. From Ref. [JANUS, 1999].

APPENDIX B. EXPERIMENTAL DESCRIPTIVE STATISTICS

This appendix contains the descriptive statistics summary of all 50 model replications for each tactical profile analyzed in Chapter 4. Column labels are defined following the tables.

NOE #1

	RU	RH	RK	BTU	ME	RE	BH	BK
Total	529	140	481	1292	813	6650	98	95
Mean	10.58	2.80	9.62	25.84	16.26	133.00	1.96	1.90
Mean S.E.	0.33	0.26	0.32	0.27	0.57	15.23	0.28	0.27
Std Dev	2.37	1.81	2.23	1.92	4.04	107.67	1.56	1.40
Median	11.0	3.0	9.0	25.5	17.0	100.0	2.0	2.0
High	16	8	15	31	25	450	8	6
Low	6	0	5	22	9	0	0	0

NOE #2

	RU	RH	RK	BTU	ME	RE	BH	BK
Total	435	178	578	563	1000	6600	69	81
Mean	8.70	3.56	11.56	11.26	20.00	132.00	1.38	1.62
Mean S.E.	0.40	0.28	0.38	0.21	0.63	15.89	0.20	0.23
Std Dev	2.84	1.98	2.67	1.51	4.44	112.38	1.28	1.26
Median	9.0	3.5	12.0	11.0	20.5	100.0	1.0	1.0
High	17	8	16	15	30	450	6	5
Low	4	0	5	9	10	0	0	0

Contour #1								
	RU	RH	RK	BTU	ME	RE	BH	BK
Total	289	222	711	397	1248	5400	71	87
Mean	5.78	4.44	14.22	7.94	24.96	108.00	1.42	1.74
Mean S.E.	0.39	0.33	0.39	0.29	0.82	14.73	0.20	0.25
Std Dev	2.78	2.34	2.78	1.92	5.84	104.18	1.39	1.59
Median	6.0	4.0	14.0	7.5	25.0	50.0	1.0	1.0
High	11	9	20	14	39	400	6	7
Low	0	0	9	6	16	0	0	0

Contour #2								
	RU	RH	RK	BTU	ME	RE	BH	BK
Total	360	210	632	292	1122	5700	81	80
Mean	7.20	4.20	12.64	5.84	22.44	114.00	1.62	1.60
Mean S.E.	0.40	0.30	0.39	0.29	0.84	15.05	0.23	0.22
Std Dev	2.84	2.10	2.78	2.03	5.90	106.45	1.46	1.41
Median	7.0	4.0	13.0	5.0	23.0	100.0	1.0	1.0
High	13	12	18	12	36	450	5	5
Low	2	1	7	3	9	0	0	0

Low Level								
	RU	RH	RK	BTU	ME	RE	BH	BK
Total	667	84	305	183	470	5350	66	78
Mean	13.34	1.68	6.10	3.66	9.40	107.00	1.32	1.56
Mean S.E.	0.57	0.25	0.49	0.26	0.79	14.43	0.18	0.21
Std Dev	4.03	1.77	3.45	1.81	5.61	102.03	1.33	1.49
Median	13.0	1.0	6.5	3.5	10.0	100.0	1.0	1.0
High	20	6	13	8	23	350	7	6
Low	6	0	0	1	0	0	0	0

NOE #3								
	RU	RH	RK	BTU	ME	RE	BH	BK
Total	402	214	607	592	1089	7750	70	69
Mean	8.04	4.28	12.14	11.84	21.78	155.00	1.40	1.38
Mean S.E.	0.33	0.37	0.31	0.26	0.67	19.18	0.20	0.20
Std Dev	2.32	2.62	2.16	1.83	4.73	135.62	1.32	1.69
Median	8.0	3.0	12.0	11.0	20.0	100.0	1.0	1.0
High	12	10	17	18	33	600	4	7
Low	3	0	8	10	14	0	0	0

NOE #4								
	RU	RH	RK	BTU	ME	RE	BH	BK
Total	377	196	632	413	1135	7250	54	64
Mean	7.54	3.92	12.64	8.26	22.70	145.00	1.08	1.28
Mean S.E.	0.31	0.29	0.30	0.17	0.63	15.73	0.15	0.18
Std Dev	2.22	2.07	2.15	1.17	4.46	111.23	1.03	1.11
Median	7.0	4.0	13.0	8.0	22.0	150.0	1.0	1.0
High	14	9	17	11	36	450	3	5
Low	3	1	7	7	16	0	0	0

NOE #5								
	RU	RH	RK	BTU	ME	RE	BH	BK
Total	355	206	649	352	1123	7400	68	103
Mean	7.10	4.12	12.98	7.04	22.46	148.00	1.36	2.06
Mean S.E.	0.33	0.34	0.32	0.25	0.70	13.99	0.19	0.29
Std Dev	2.31	2.38	2.26	1.77	4.94	98.95	1.37	1.25
Median	7.0	4.0	13.0	6.5	22.0	150.0	1.0	2.0
High	13	13	18	12	38	400	5	4
Low	2	1	8	5	11	0	0	0

Table Column Label Definitions:

- RU: Undetected Red agents
- RH: Red agents Hit
- RK: Red agents Killed
- BTU: Blue agent Teams Utilized (tandem teams)
- ME: Missiles Expended
- RE: Rounds Expended
- BH: Blue agents Hit (single helicopters)
- BK: Blue agents Killed (single helicopters)

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

Comanche Program Manager, Department of the Army, *Comanche Pre-Production Prototype (PPP) Program; Test & Evaluation Master Plan (TEMP)*, 1999.

Crouch, T., "The RAH-66 Comanche: Key to the Army Vision", *Army Aviation*, March/April, pp. 18-24, Westport, CT: Army Aviation Publications, Inc., 2000.

Department of Defense, *Mandatory Procedures for Major Defense Acquisition Programs and Major Automated Information Systems (DoD 5000.2-R)*, Washington, DC: U.S. Government Printing Office, 1996.

Ferber, J., *Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence*, London: Addison-Wesley, 1999.

Headquarters, Department of the Army, *Aircrew Training Manual Attack Helicopter, AH-64* (Training Circular 1-214), Washington, DC: U.S. Government Printing Office, 1992.

Headquarters, Department of the Army, *Attack Helicopter Operations* (Field Manual 1-112), Washington, DC: U.S. Government Printing Office, 1997.

Headquarters, Department of the Army, *Cavalry Operations* (Field Manual 17-95), Washington, DC: U.S. Government Printing Office, 1996.

Headquarters, Department of the Army, *Cavalry Troop* (Field Manual 17-97), Washington, DC: U.S. Government Printing Office, 1995.

Headquarters, Department of the Army, *Infantry-Based Opposing Force* (Field Manual 100-63), Washington, DC: U.S. Government Printing Office, 1996.

Headquarters, Department of the Army, *Operational Requirements Document for the RAH-66 Comanche Armed Reconnaissance/Attack Helicopter* (ORD), Washington, DC: U.S. Government Printing Office, 1999.

Headquarters, Department of the Army, *Tactics, Techniques and Procedures for the Regimental Aviation Squadron* (Field Manual 1-114), Washington, DC: U.S. Government Printing Office, 2000.

Headquarters, Department of the Army, *Technical Manual Operator's Manual for Army OH-58D Helicopter,* (Technical Manual 55-1520-248-10), Washington, DC: U.S. Government Printing Office, 1992.

Headquarters, Department of the Army, *The Soviet Army: Troops, Organization, and Equipment* (Field Manual 100-2-3), Washington, DC: U.S. Government Printing Office, 1991.

Holland, J. H., *Hidden Order: How Adaptation Builds Complexity*, Reading, MA: Addison-Wesley, 1995.

Ilachinski, A., Center for Naval Analyses, *ISAAC Irreducible Semi-Autonomous Adaptive Combat: An Artificial-Life Approach to Land Warfare*, http://www.cna.org/isaac/, 24 July 00.

Ilachinski, A., *Irreducible Semi-Autonomous Adaptive Combat (ISAAC): An Artificial-Life Approach to Land Warfare (U)*, (Center for Naval Analyses Research Memorandum CRM 97-61.10), Alexandria, VA: Center for Naval Analyses, 1997.

Resnick, M., *Turtles, Termites, and Traffic Jams: Explorations in Massively Parallel Microworlds*, Cambridge, MA: MIT Press, 1997.

Schmidt, K., *Modeling and Simulation Support for the Operational Test and Evaluation of a Tactical Airborne Reconnaissance System*, Master's thesis, Naval Postgraduate School, Monterey, CA, 2000.

Stine, J., *Representing Tactical Land Navigation Expertise*, Master's thesis, Naval Postgraduate School, Monterey, CA, 2000.

U.S. Army Training and Doctrine Command Analysis Center (TRAC), *JANUS version 7.06D*, [Computer Software], White Sands Missile Range, NM, 1999.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center 2
   8725 John J. Kingman Road, STE 0944
   Ft. Belvoir, Virginia 22060-6218

2. Dudley Knox Library 2
   Naval Postgraduate School
   411 Dyer Road
   Monterey, California 93943-5101

3. Dr. Donald Gaver 2
   Operations Research Dept.
   Naval Postgraduate School
   Monterey, California 93943-5221

4. Dr. Patricia A. Jacobs 1
   Operations Research Dept.
   Naval Postgraduate School
   Monterey, California 93943-5221

5. Mr. John Hiles 1
   MOVES Academic Group
   Naval Postgraduate School
   Monterey, California 93943-5118

6. MAJ Thom Crouch 1
   Systems Management Dept.
   Naval Postgraduate School
   Monterey, California 93943-5001

7. CPT Craig S. Unrath 1
   5009 James Loop
   Killeen, Texas 76542

8. COL Wayland E. Parker 1
   Military Assistant for Army Aviation Systems
   Room 1C730
   The Pentagon
   Washington, District of Columbia 20301-1700

9. Dr. Alfred Brandstein 1
MCCDC (C 45)
Studies and Analysis Division
3300 Russell Road
Quantico, Virginia 22134-5001

10. Dr. Rudolph P. Darken 1
MOVES Academic Associate
Code CS/DR
Naval Postgraduate School
Monterey, California 93943

11. CDR Art Galpin 1
Joint Special Operations Simulation Office
USSOCOM (SORR-SCS)
MacDill AFB, Florida 33621

12. Mr. Paul Bloch 1
The Boeing Company
PO Box 3707
85-80
Seattle, Washington 98124-2207

13. Mr. Hall 1
Comanche TSM
ATTN: ATZQ-CDC-T
Fort Rucker, Alabama 36362-5000

14. Mr. Ferguson 1
U.S. Army ATTC
ATTN: STEAT-TS-P
Fort Rucker, Alabama 36362

15. Mr. Perez 1
U.S. Army OTC
ATTN: CSTE-TAV
Fort Hood, Texas 76544

16. MAJ Grimm 1
U.S. Army TEMA, ATTN: DAC-TE
Chief of Staff
200 Army Pentagon
Washington, District of Columbia 20310-0120

17. Mr. Madron 1
U.S. Army TECOM
ATTN: AMSTE-TM-L
Aberdeen Proving Ground, Maryland 21005

18. Mr. Ralph Redman 1
U.S. Army OEC
ATTN: CSTE-EAV
4501 Ford Avenue
Alexandria, Virginia 22302-1458

19. Mr. Walsh 1
U.S. Army TRADOC
ATTN: ATCD-MV
Fort Monroe, Virginia 23651

20. Mr. Donahue 1
U.S. Army STRICOM
ATTN: AMCPM-ACTS
12750 Research Parkway
Orlando, Florida 32826

21. Mr. Brian Smith 1
U.S. Army Research Laboratory
ATTN: AMSRL-SL-I
Aberdeen Proving Ground, Maryland 21005-5068

22. Mr. Sterling B. Crawford 1
SAIC
1755 Jeff Davis Highway
McLean, Virginia 22102

23. Mr. Brent Crabtree 1
IDA
1801 North Beaurgard Street
Alexandria, Virginia 22302

24. Mr. Ronald Scotka 1
TWR
1801 North Beaurgard Street
Alexandria, Virginia 22302